Muscle activation patterns during level walking and stair ambulation

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 17 April 2026, accepted 27 April 2026, date of publication 30 April 2026, date of current version 19 May 2026.
Digital Object Identifier 10.1109/ACCESS.2026.3689184

RESEARCH ARTICLE

Toward Automated Co-Simulation of Multi-Energy-Systems: An Ontology-Driven Approach

MATIA TORLINI[1], PIETRO RANDO MAZZARINO[1], (Member, IEEE), DANIELE SALVATORE SCHIERA[1], (Member, IEEE), JULIÁN ARENAS-GUERRERO[2], LUCA BARBIERATO[1], (Member, IEEE), LORENZO BOTTACCIOLI[1], (Member, IEEE), AND EDOARDO PATTI[1], (Member, IEEE)
[1]Politecnico di Torino, Turin, Italy
[2]Universidad Politécnica de Madrid, Madrid, Spain

Corresponding author: Lorenzo Bottaccioli (lorenzo.bottaccioli@polito.it)

This work was supported by the Co-Simulation of Multi-Energy systems for Energy Transition (COMET), Italian National funded project under PNRR M4C2, Investimento 1.4-Avviso n. 3138 del 16/12/2021-National Centre for HPC, Big Data and Quantum Computing (HPC) under Grant CN00000013.

ABSTRACT Multi-Energy-System (MES) integrates diverse energy carriers, digital networks, and human interactions, requiring accurate simulations for efficient urban energy management. Coupled Simulation (Co-Simulation) enables complex systems like MESs to be simulated as a whole by linking and coordinating their individual sub-modules. Current co-simulation approaches face major challenges: connecting heterogeneous models and preparing input data often demand significant manual effort and domain expertise. Most frameworks provide technical interoperability but cannot fully capture each model's data requirements and interfaces, limiting automation, scalability, and reproducibility. To address these limitations, we present COSIMO, a framework that organizes models, technologies, parameters, and their interconnections in a structured format that software tools can interpret automatically. COSIMO streamlines the creation and management of co-simulation setups, supporting scalable and reproducible urban energy simulations. We demonstrate its effectiveness through scenarios from neighborhood to large urban districts—including photovoltaic systems, battery storage, and occupancy-driven loads—showing improved interoperability, reproducibility, and seamless integration of diverse models and data.

INDEX TERMS Multi-energy-systems, co-simulation, ontology, knowledge graph, semantic interoperability, automation.

ACRONYMS

BEO	Building Elements Ontology.
ETSI	European Telecommunications Standards Institute.
FMI	Functional Mock-up Interface.
FMU	Functional Mock-up Unit.
GOPPRR	Graph, Object, Port, Property, Role, Relationship.
HELICS	Hierarchical Engine for Large Infrastructure Co-simulation.
IFC	Industry Foundation Classes.
LoD	Level of detail.
MES	Multi-Energy-System.
OWL	Web Ontology Language.
PV	Photovoltaic.
RDF	Resource Description Framework.
RDFS	RDF Schema.
RML	RDF Mapping Language.
S4S	SAREF for Systems.
SAREF	Smart-Appliances REFerence Ontology.

The associate editor coordinating the review of this manuscript and approving it for publication was Wencong Su.

SD	Software Description Ontology.
SDM	Software Description Ontology for Models.
SPARQL	SPARQL Protocol and RDF Query Language.
UEM	Urban-Energy-Modelling.

I. INTRODUCTION

To meet the Paris Agreement goals and national decarbonization targets, many countries are transforming their urban energy infrastructures, integrating electricity, heating, and cooling networks with digital control systems [45]. Achieving the full potential of these systems requires coordinated planning and operation of energy networks, which have historically been designed and managed in isolation. This coordination is reshaping urban infrastructures from single-carrier networks into complex, interconnected Multi-Energy-System (MES) [13]. As these systems become more interconnected, accurately modelling and simulating their behaviour becomes increasingly complex, revealing the limitations of traditional monolithic simulation tools, which also struggle to scale to the urban level, where hundreds or thousands of interconnected assets must be simulated simultaneously. Additionally, monolithic models are challenging to adapt to new contexts, thereby limiting their utility for planning and operational decision-making. Addressing these limitations requires a modelling approach that coordinates multiple specialised tools, maintaining consistent data exchange and synchronised execution across interdependent models.

A. INTEGRATED CO-SIMULATION AND SEMANTIC WEB TECHNOLOGIES

Analysing urban energy systems requires understanding complex interactions across multiple scales and domains. Consider a neighbourhood integrating renewable energy through rooftop Photovoltaic (PV) systems, heat pumps, and district heating networks. Such analysis demands the coordination of specialised simulators, including solar generation models that predict PV output from weather data and panel orientation, building thermal models that calculate heating demands, heat pump controllers that optimise operation modes, thermal storage systems that manage charge/discharge cycles, and district heating network simulators that handle fluid dynamics. Each operates at distinct temporal scales — ranging from seconds for power electronics to minutes for thermal dynamics and hours for energy pricing — while interactions span multiple energy carriers and spatial boundaries.

To address this complexity, coupled simulation (co-simulation) enables global simulation of complex systems through the composition of specialised simulators. Each simulator acts as a black box, consuming inputs and producing outputs while encapsulating its internal behaviour. This approach preserves intellectual property, enables the distribution of computational loads, and allows for model reuse across different studies [12], [19]. As cyber-physical systems grow increasingly complex, such integration becomes essential to capture the intricate interactions between energy generation, conversion, storage, and consumption.

Various co-simulation frameworks have emerged to manage communication and synchronisation among heterogeneous components, typically employing publish-subscribe patterns or master-slave coordination architectures to enable scalable and flexible integration [9]. However, these frameworks primarily address syntactic integration, ensuring compatible data formats and communication protocols, while leaving semantic interoperability unresolved.

This lack of semantic interoperability poses significant integration challenges despite successful data exchange. When a building model outputs *thermal power demand*, one simulator might interpret this as instantaneous power in kilowatts, another as energy consumption in kilowatt-hours, and a third as peak capacity. Unit mismatches compound these issues: temperature values exchanged in Celsius while another simulator expects Kelvin, or thermal energy flows provided in British thermal units per hour instead of watts. Such ambiguities can lead to integration errors that undermine simulation validity.

Beyond semantic challenges, co-simulation construction at the urban scale requires gathering and preparing heterogeneous input data for diverse models — a task typically performed manually, with significant time investment and error risk. This may require manually collecting building geometry, thermal properties, occupancy schedules, PV panel specifications, solar irradiance data, heat pump performance curves, thermal storage capacities, district heating network topology, etc.

To address these semantic interoperability and data integration challenges, ontologies and Semantic Web technologies offer solutions through flexible, interoperable data structures. Ontologies provide *"a formal, explicit, shared specification of a conceptualisation"* [15], enabling formal representation of domain concepts and relationships. These technologies build upon a layered architecture. The Resource Description Framework (RDF) [46] provides a graph-based data model encoding information as *subject-predicate-object* triples, enabling flexible data linking. RDF Schema (RDFS) [47] adds semantic constructs such as class hierarchies. The Web Ontology Language (OWL) [48] enables advanced reasoning through Description Logic. SPARQL Protocol and RDF Query Language (SPARQL) [49] provides pattern matching and querying capabilities, including federated queries across distributed datasets [32]. The RDF Mapping Language (RML) [20] enables automated transformation of diverse data formats into RDF triples, facilitating the construction of semantically rich knowledge graphs from the fragmented datasets typical of urban energy systems.

These technologies underpin knowledge graphs, which organise information as interconnected entities with formal semantics. In urban energy co-simulation, an ontology could formally define that *thermal power demand* represents energy per unit time in watts, with explicit relationships to the generating building component, time step, and target temperature, eliminating current ambiguities. Such formal definitions enable automated validation, unit conversion, and semantic consistency checking across the entire co-simulation framework.

B. SCOPE AND CONTRIBUTIONS

Motivated by the challenges of co-simulating urban MESs at scale and the potential of ontological approaches, this work extends [44] by investigating how semantic methods can enable automated, interoperable, and reproducible co-simulation of urban energy systems.

While [44] introduced a preliminary ontology for semantically describing simulation units and demonstrated feasibility through a single building validation case, the present work takes this foundation considerably further. We transition from conceptual validation to a full framework called COSIMO, which uses three integrated knowledge graphs (Model Catalog, Topologies Catalog, and Data Graph) to automate the entire process from heterogeneous urban data to executable co-simulations. We also ground our approach in established semantic web standards rather than developing abstractions from scratch, which improves both expressiveness and interoperability with existing tools. Beyond the basic model descriptions in [44], we introduce a Topologies Catalog that captures how models connect and synchronise with each other, allowing the same model to work in different coupling scenarios with appropriate timing configurations. Additionally, we address data integration more systematically by using declarative mappings and model generation services to bridge the gap between diverse urban datasets and simulation requirements.

Specifically, it focuses on four interrelated research questions, each of which motivates one or more concrete requirements of the COSIMO framework, formally derived through the gap analysis presented in Section II:

- **RQ1: Semantic Representation** – How can urban data and simulation models be formally represented in a unified, machine-readable format?
- **RQ2: Model Coupling** – How can semantic annotations and scenario descriptions enable the automatic instantiation and interconnection of simulation units, reducing manual setup effort?
- **RQ3: Parameters Mapping and Reproducibility** - How can heterogeneous urban data sources be systematically mapped to model parameters, ensuring consistent and reproducible simulation results?
- **RQ4: Extensibility and Reusability** – How can semantic representations support the addition of new simulation models and the reuse of existing semantic descriptions across different urban energy scenarios?

Guided by the research questions, we developed COSIMO, an ontology-based framework that, to the best of our knowledge, provides the first end-to-end semantic approach for urban multi-energy system co-simulation. This framework systematically connects heterogeneous data sources to executable simulation configurations through formal ontological representations. While existing work at the intersection of energy simulation and ontologies primarily focuses on device modelling, building semantics, smart home reasoning, or digital-twin preparation, the automated generation of executable co-simulation configurations remains largely unaddressed. This work therefore contributes not simply the adoption of established ontologies, but their systematic integration into a unified semantic framework capable of automating the end-to-end construction of urban multi-energy system co-simulations.

The rest of this manuscript is organised as follows: Section II reviews related work on semantic urban energy modelling and application of ontologies in co-simulation contexts, highlighting identified gaps. Section III presents COSIMO's architecture in detail. Section IV illustrates the methodology through a representative urban energy use case, including automated generation of co-simulation setups. Section V discusses evaluation metrics, experimental results, and implications for scalability, reproducibility, and interoperability. Section VI discusses the key remarks and limitations of the proposed solution. Finally, Section VII concludes the paper and outlines directions for future research.

II. RELATED WORK

A. URBAN ENERGY MODELLING AND SEMANTIC FOUNDATIONS

Urban-Energy-Modelling (UEM) relies on consistent, structured representations of buildings and energy-related data. Standards such as the Common Information Model, Industry Foundation Classes (IFC) [6], and CityGML with its Application Domain Extensions [14] offer unified models for describing energy systems and properties across the urban scale. However, in practice, urban energy data are scattered across heterogeneous formats and databases, each encoding different semantic information. This fragmentation impedes seamless interoperability, particularly as urban energy systems grow in scale and complexity. To overcome these challenges, knowledge graphs have emerged as a promising approach [17], providing a unifying semantic layer that can federate disparate data sources, enable flexible information access, and serve as an integrative infrastructure to bridge fragmented standards. Smart-Appliances REFerence Ontology (SAREF) [?]db@?, developed by European Telecommunications Standards Institute (ETSI), defines core concepts for smart appliance representation alongside modular extensions addressing specific domains. Beyond serving as a general semantic foundation, the SAREF ecosystem has also been progressively adopted in application-oriented smart environments. GarcÍa-Castro et al. [10] document the evolution of SAREF from a core ontology for smart appliances into a modular ecosystem of extensions supporting interoperability across multiple domains. In parallel, Reda et al. [33] show how SAREF-based semantic representations can support rule-based reasoning in smart home scenarios through OWL and SWRL. These works confirm the suitability of SAREF as a common semantic layer for interoperable energy-related applications. However, their primary focus remains device interoperability, semantic reasoning, and application support, rather than the automated composition of executable co-simulation scenarios. This distinction is particularly relevant for urban multi-energy systems, where semantic representations must support not only domain interoperability, but also model selection, parameter preparation, and variables coupling. In particular, SAREF4BLDG [31] incorporates relevant subsets of the IFC standard for modelling building devices and equipment, while SAREF4SYST [23] provides patterns for representing systems and their interconnections. This high-level perspective complements the Building Elements Ontology (BEO) [30], which provides IFC-derived classes for specific building elements such as rooftops. Together with GeoSPARQL [3], the de facto standard for georeferenced data representation and querying, the SAREF suite and BEO form an integrated semantic framework spanning building devices, spatial topology, physical elements, and geographic context. The potential of such semantic representation to automate simulation workflows has been demonstrated in recent works: Huang et al. [18] evaluated knowledge graph-based integration of CityGML and IFC data for solar energy simulations, comparing traditional data integration pipelines against ontology-driven approaches. Their feasibility study demonstrated that semantic linking enables unified SPARQL querying across heterogeneous building and geospatial data sources while preserving original formats. Wu et al. [50] proposed an ontology-based automatic framework that integrates multiple data sources, including weather, building geometry, and internal heat gains, incorporating rule-based reasoning for automatic thermal zoning and instance-based mapping for generating building simulation models. While these efforts demonstrate the potential of semantic web technologies for integrating heterogeneous data, applications at the urban scale require a comprehensive framework that unifies the representation of simulation models, their configurations, and the mechanisms for transforming heterogeneous data into tool-specific formats.

B. ONTOLOGIES SUPPORTING CO-SIMULATION

The fundamental challenge in co-simulation is establishing semantic interoperability among models developed independently using different tools and conventions. Early recognition of this problem led to proposals for semantically richer descriptions of model interfaces beyond the primitive types provided by standards like Functional Mock-up Interface (FMI). FMUont [26] provides an OWL-based ontology for Functional Mock-up Unit (FMU), defining classes such as FMU, hasInputVariable, hasOutputVariable, hasParameter, Medium, Quantity, Unit, and SemanticType to enable unambiguous semantic description of model interfaces and automated topology derivation.

Rindarøy et al. [36] present OSP-IS, an ontology to enrich FMI simulation models by adding human and machine-readable semantics. OSP-IS enforces validation rules based on meaningful physical exchanges, while violations are classified into error types, providing clear feedback, and ensuring valid connections.

Li et al. [24] extend FMU-level descriptions to complete co-simulation scenarios through a Graph, Object, Port, Property, Role, Relationship (GOPPRR)-based ontology (Graph, Object, Port, Property, Role, Relationship) that captures design and timing information, topological relationships, and simulation parameters. While GOPPRR formalises system integration patterns, the approach primarily addresses topological aspects, with domain-specific physical semantics remaining implicit within individual models.

The FMI-ontology suite [42], [43] addresses the connection between simulation models and domain knowledge descriptions, directly linking real entities to models. The FMI-ontology provides an RDF/OWL transcription of FMI standard definitions, while the SMS-ontology (Systems, Models, Simulations) establishes a formalism-independent framework connecting real or conceptual systems to their abstract representations, demonstrating how ontologies can connect domain semantics with simulation model descriptions, though the practical application to automated co-simulation setup from real-world data sources remains to be investigated.

In the context of planning and evaluating energy co-simulation scenarios involving experts from diverse domains, Schwarz et al. [37], [38] propose the use of a semantic media-wiki of components from where to retrieve simulation units starting from the definition of a co-simulation scenario in the form of a mind map. Moreover, external ontological knowledge is linked to the elements in the mind map, exploiting the meaning defined there for the entities of interest.

According to [41], automatic integration of models in the mobility sector can be achieved by leveraging ontologies for two purposes: establishing standardised terminology and creating mappings that link ontological concepts to model metadata.

Novák and Šindelář [28] develop a semantic-based approach that partially automates the construction of simulation models for industrial facilities. Their system encodes knowledge about both plants and simulations, then uses SPARQL-based retrieval to match plant components with corresponding models from a repository.

The reviewed works served as direct inputs to COSIMO's design, both methodologically and in terms of gap identification. From a methodological standpoint, Wu et al. [50] directly inspired our use of RML mappings and model generation services within the Data Graph pipeline, demonstrating how ontology-driven transformations can bridge heterogeneous urban data sources and simulator-specific formats. FMUont [26] and the SMS-FMI suite [42], [43] informed the design of the Model Catalog, particularly the formal description of model interfaces and their linking to domain knowledge through RDF/OWL representations. The GOPPRR-based formalisation of Li et al. [24] directly inspired the structure of the Topologies Catalog, where coupling patterns and timing configurations are persisted as reusable templates, while Novák and Šindelář [28] motivated our SPARQL-based composition workflow for matching physical system components to simulation models. From the perspective of gap identification, Schwarz et al. [37], [38] highlighted the need for assisted scenario construction in multi-domain co-simulation contexts, directly motivating RQ2 on automated model coupling, while OSP-IS [36] underlined the importance of connection validation beyond syntactic compatibility. More broadly, the fragmentation observed across all reviewed contributions, each one addressing isolated aspects of the co-simulation lifecycle without providing an end-to-end workflow, collectively motivated the formulation of requirements presented in the following section.

C. GAP ANALYSIS

The literature examined reveals that existing semantic and co-simulation approaches address important but isolated aspects of the co-simulation construction problem. While each contribution provides valuable solutions to specific challenges, whether by describing interface semantics [26], modelling coupling errors [36], topological relationships [24], or integrating domain knowledge [37], [50], they remain fundamentally fragmented. None offers a comprehensive framework that integrates these complementary elements into a unified workflow capable of addressing the complete co-simulation lifecycle from real-world data to executable co-simulations. Table 1 synthesises these works according to the requirements presented in the following paragraph. To enable the construction of co-simulation at scale, a comprehensive framework must fulfil the following requirements, each addressing critical gaps in existing approaches and motivated by the research questions RQs defined in Section I-B:

- **R1 - Unified Semantic Representation**: provide machine-readable representation of urban data and simulation models through shared ontological foundations, addressing RQ1. None of the reviewed works simultaneously represents both urban data and simulation models under a common semantic layer, with existing contributions addressing either aspect in isolation. A shared ontological foundation is therefore identified as a prerequisite for enabling automated reasoning across heterogeneous urban assets and simulation components.
- **R2 - Semantic Connection Validation**: ensure physical meaningfulness and engineering correctness of model connections beyond syntactic compatibility, addressing RQ2. Existing works validate connections syntactically but fail to ensure physical meaningfulness, leaving automatic and correct instantiation of simulation units unaddressed. R2 directly operationalizes RQ2 by defining connection validation as a necessary condition for reliable automated model coupling, as highlighted in [36], [37], and [38].
- **R3 - Topology Preservation**: maintain and persist coupling topologies, connection patterns, and inter-model relationships for reusability across simulation scenarios, addressing RQ2 and RQ4. Existing works formalise topological structures without persisting them as reusable artifacts, requiring redefinition for each new scenario. R3 operationalizes both RQ2 and RQ4 by establishing topology persistence as the mechanism enabling automatic model interconnection and semantic reusability across diverse urban energy scenarios.
- **R4 - Context-Dependent Coupling and Synchronisation**: dynamically configure temporal synchronisation strategies and coupling patterns based on the specific combination of models present in each co-simulation topology, addressing RQ2. No reviewed work configures synchronisation strategies dynamically based on the specific model combination instantiated, leaving temporal coordination an unresolved challenge for automatic model coupling. R4 operationalizes RQ2 by establishing context-dependent synchronisation as a necessary condition for correctly coordinating models operating at different temporal scales upon automatic instantiation.
- **R5 - Automated Parameter Mapping**: support systematic transformation from heterogeneous datasets to executable models at scale, addressing RQ3. Existing works either perform parameter mapping manually or address it only partially, without providing a systematic and reproducible pipeline from heterogeneous urban data sources to simulator-specific formats. R5 operationalizes RQ3 by establishing automated parameter transformation as the concrete mechanism through which consistent and reproducible simulation configurations are achieved at scale.
- **R6 - Extensible Architecture**: enables the integration of new models and the reuse of semantic descriptions without requiring workflow redesign, addressing RQ4. Existing frameworks are tightly coupled to specific tools or domain assumptions, requiring significant redesign effort when new models or scenarios are introduced. R6 operationalizes RQ4 by establishing architectural extensibility as the direct technical enabler of semantic reusability, ensuring that new simulation models can be integrated and existing descriptions reused across diverse urban energy scenarios without disrupting the overall workflow.

In particular, these requirements collectively define the foundation for a semantically-grounded approach to co-simulation construction. While no existing work satisfies all requirements simultaneously, their partial fulfilment across different frameworks demonstrates the feasibility of individual components. In particular, Table 1 synthesises the reviewed contributions against the six identified requirements, where fully supported (✓) indicates comprehensive coverage, partially supported (∼) indicates that relevant mechanisms are present but incomplete or limited in scope, and not supported (×) indicates no coverage. The table reveals a consistent pattern of fragmentation. Wu et al. [50] address R1 and R5 but omit model coupling and topology management. FMUont [26] supports R2 and R6 but not urban data representation or parameter mapping. OSP-IS [36] fully supports R2 and partially R3 – it classifies coupling errors but does not persist topology patterns as reusable templates. Li et al. [24] fully supports R3 but only partially R6, as its abstractions offer limited reusability without a fully extensible architecture. SMS-FMI [42], [43] partially addresses R1 by linking domain knowledge to models, without supporting automated scenario construction. Schwarz et al. [37], [38] fully support R1 and partially R2 – ontological knowledge is linked to scenario elements but connection validation remains informal. Novák and Šindelář [28] and Stepien et al. [42] partially contribute to R1 and R5 within narrow domain scopes without generalising to a complete workflow. The proposed COSIMO framework addresses this gap by simultaneously satisfying requirements R1 to R6 through an integrated semantic approach that brings these capabilities into a coherent methodology, thereby constituting the principal contribution and novelty of this work.

TABLE 1. Comparison of semantic co-simulation approaches against identified requirements (✓ fully supported, × not supported, ∼ partially supported).

Work	R1	R2	R3	R4	R5	R6	Primary Focus
Wu et al. [50]	✓	×	×	×	✓	∼	Transformation of semantic data into simulator-specific formats
FMUont [26]	×	✓	∼	×	×	✓	Semantic description of FMU interfaces
OSP-IS [36]	×	✓	∼	×	×	∼	Connection errors classification
Li et al. [24]	×	×	✓	×	×	∼	Formalisation of co-simulation topology
SMS-FMI [42], [43]	∼	×	×	×	×	∼	Linking domain knowledge to simulation models
Schwarz et al. [37], [38]	✓	∼	×	×	×	∼	Semantic wiki for collaborative scenario definition
Stepien et al. [41]	∼	×	×	×	✓	∼	Ontology-based parameter transformation
Novak et al. [28]	✓	∼	×	×	×	∼	SPARQL-based component-to-model matching
Proposed Work	✓	✓	✓	✓	✓	✓	**Comprehensive framework for automated co-simulation construction**

III. METHODOLOGY

This section presents COSIMO's architecture and demonstrates how semantic web technologies enable automated construction of co-simulation environments for urban multi-energy systems. To address the challenges presented in the previous section,

COSIMO leverages three independent knowledge management activities: i) gathering scenario data and definition, ii) cataloguing simulation models, and iii) defining coupling topologies. By formalising this knowledge in a machine-readable way, the framework enables automated reasoning across these distinct knowledge sources.

A. HELICS FOR CO-SIMULATION

To implement the co-simulation capabilities required by our framework, we selected Hierarchical Engine for Large Infrastructure Co-simulation (HELICS) [16] as the underlying communication and synchronisation infrastructure. HELICS encapsulates each simulation component as a federate, a software entity responsible for managing time advancement and data exchange for its underlying model. Federates communicate through a publish-subscribe mechanism [9], with publications and subscriptions declared in configuration files. This declarative approach aligns well with our semantic methodology, as it separates model logic from communication specifications and enables automated generation of federate configurations.

This framework provides flexible time-advancement mechanisms and robust synchronisation strategies that maintain temporal consistency across federates operating at different time scales. The numerical stability, coupling consistency, and computational efficiency of HELICS have been thoroughly validated in the literature: comparative studies [2], [40] have demonstrated that dedicated co-simulation orchestration frameworks ensure tight temporal coupling and numerically stable data exchange across heterogeneous federates, making them particularly well-suited for complex, large-scale scenarios. Recent work [7] has further corroborated the maturity of such infrastructures under demanding deployment and scalability conditions. The present framework, therefore, builds on a numerically validated and coupling-proven orchestration foundation, with its contribution lying not in the co-simulation engine itself, but at the scenario-construction level, ensuring that federate configurations are semantically coherent.

B. ARCHITECTURAL OVERVIEW

COSIMO's architecture separates concerns into three main areas that collectively capture all information required

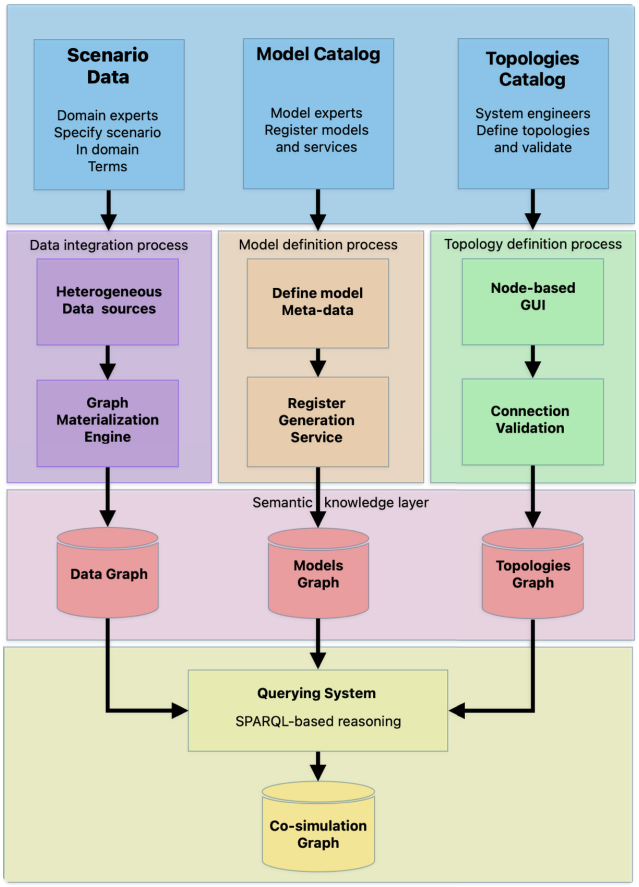

FIGURE 1. General architecture of the semantic framework workflow showing the transformation of domain expert specifications into executable co-simulations through integrated knowledge graphs.

for automated co-simulation construction (see Figure 1). These graphs can be maintained independently by different stakeholders, yet queried together to synthesise complete simulation environments:

i) The *Data Graph* is a semantic representation of the physical assets being simulated, integrating information from diverse sources.

ii) The *Model Catalog* is a registry of available simulation components equipped with an HELICS interface, documenting their input/outputs, requirements, and the data transformation services needed to prepare inputs in simulator-specific formats.

iii) The *Topologies Catalog* is a library of validated coupling patterns that define how models can be correctly connected and synchronised.

These three graphs are queried together to identify compatible models, instantiate and interconnect them, and generate the configuration files required by each one. The methodology unfolds through five connected stages. First, we explain how simulation models are catalogued with formal interface descriptions, requirements, and links to domain ontologies that provide semantic foundations for scenario interpretation. Second, we describe how the Model Catalog supports the assisted construction of valid coupling patterns and how timing rules are captured in the topologies graph for reuse. Third, we explain how the data graph integrates heterogeneous data sources and supports declarative scenario specification. Fourth, we show how the querying system exploits the three graphs to invoke model-generation services and synthesise complete co-simulation configurations. Finally, we explain how the co-simulation configuration translates into instantiated models with automatically configured HELICS interfaces that are correctly connected and synchronised according to their context. This architecture enables domain experts to specify multi-energy scenarios declaratively, leaving the technical complexities of co-simulation construction and validation to the framework.

C. MODEL CATALOG

The Model Catalog (see Figure 1) constitutes a machine-readable registry of models equipped with an HELICS interface participating in co-simulation workflows. Anchoring this catalog in formal ontologies facilitates reasoning about each component's data. To represent simulation software and associated services, we adopt the Software Description Ontology (SD) [21] and its extension, the Software Description Ontology for Models (SDM) [11]. SD was selected for its generality: unlike ontologies focused exclusively on simulation or scientific computing, SD maintains sufficient breadth to represent both simulation models and auxiliary software components participating in co-simulation workflows, including model generation services and data processing utilities. SD provides foundational concepts for capturing software metadata, including name, version, creator, and execution constraints.

SDM extends this foundation with constructs specific to simulation modelling. It provides a framework for formally describing model inputs, outputs, parameters, and the internal timestep by which the model advances time.

While this framework is general, a detailed semantic representation of variables is necessary for interoperability. Variable definitions are standardized through several components: (i) links to controlled vocabulary terms defining standard variables such as "Global solar irradiance on earth surface" or "Outside dry bulb temperature", (ii) units of measure formalized using the Ontology of Units of Measurement (OM) [35], (iii) datatypes specifying whether values are numeric (float, integer), boolean, string, or array format, and (iv) value constraints defining admissible ranges and default values. These formal specifications serve a dual purpose. First, they document each model's requirements and capabilities in machine-readable form. Second, they provide the foundation for automated validation of model interconnections during topology construction. While model interfaces and parameters can be specified within the ontology, a mechanism is needed to translate data from the Data Graph into formats that specific simulation tools can interpret. To address this heterogeneity, a model may be associated with a model generation service: a software component that accepts semantic descriptions and produces executable configuration files. For example, consider a rooftop PV model generation service. This service captures the transformation logic by taking building geometry and geographic coordinates from the Data Graph, generating model configuration files with parameters such as tilt angle and rooftop slope, and outputting an executable configuration file. The unified view on urban energy data ensures that service definitions remain invariant, enabling permanent reuse across scenarios without redefinition. Generation services are themselves described using SD, enriched with semantics from the Service Ontology [8], and linked to models via the *service:consumes* property. This enables the framework to automatically identify and invoke appropriate services during the co-simulation construction process. While SD and SDM provide comprehensive descriptions of simulation models and their computational interfaces, they do not capture the relationships to the domain concepts used to express scenarios. A PV system model might be perfectly specified with all its inputs, outputs, and parameters, yet the framework still needs to know that this model is appropriate for simulating photovoltaic devices as represented in the Data Graph. The critical connection between domain knowledge in the Data Graph and simulation capabilities in the Models Graph is established through the *rdf:type* property, which links simulation models to appropriate semantic technology types. For example, the triple *catalog:PV_Model_1 rdf:type s4bldg:PhotovoltaicSystem* establishes that when a PV system appears in a scenario specification, the model *catalog:PV_Model_1* is appropriate for its simulation. This approach bridges the semantic gap between scenario entities expressed in domain terminology and computational models capable of simulating them. A minimal example of a model within the catalog is shown in Figure 2. While model specifications are currently authored using Protégé [27], more robust solutions incorporating automated validation and user-friendly interfaces should be implemented to ensure metadata quality about models.

D. TOPOLOGIES CATALOG

The Model Catalog provides complete specifications of individual simulation components, but it does not address how these components should be connected and synchronised when instantiated to form complete co-simulation systems. Questions remain about which outputs connect to which inputs and what timing constraints must be satisfied when models are instantiated. The Topologies Catalog (see Figure 1) addresses these questions by enabling the definition and validation of coupling patterns between models, which are then persisted as reusable templates. To represent system-level coupling patterns, we adopt SAREF for Systems (S4S) as a natural extension of our use of SAREF in domain modelling. This choice maintains consistency within the SAREF ecosystem while providing the necessary constructs to represent systems of systems and their compositional structures. S4S enables the representation and storage of topology configurations that capture how models are instantiated and connected within particular simulation scenarios. Importantly, a single model may participate in multiple topologies with different connection patterns and synchronisation configurations. The same model can thus be coupled differently depending on which other models it needs to interact with.

For instance, a PV model might couple to both battery and grid models in one topology, while in another topology, the same model connects only to a building electrical load without battery storage. This pattern-based approach enables significant reuse: if a district contains fifty buildings with photovoltaic systems and batteries, the same topology pattern applies to each building without requiring fifty separate configuration efforts, drastically reducing the risk of introducing coupling and timing errors. The topology serves as a template, while model instantiation creates concrete connections and parametrised executable models. The formal interface specifications established in the Model Catalog enable automated validation of proposed connections between components at topology construction time. For convenience, we developed a node-based graphical user interface to simplify the visualisation and definition of connections and timing for combinations of models within the catalog, enabling interactive creation of a new topology. When defining a new topology, each proposed connection between two models must satisfy compatibility criteria to ensure physical meaningfulness and computational correctness. The SPARQL query used to assess the validity of a connection is presented in Listing 1.

Constraints defined in the query prevent meaningless couplings from being saved in the Topologies Catalog, such as connecting temperature output to power input, catching unit mismatches such as attempting to connect watts to kilowatts without explicit conversion, identify datatype incompatibilities before runtime, and detect range mismatches that would cause inputs to consistently fall outside their valid domains.

Beyond systems connectivity, topologies encode the temporal coordination strategies required for stable co-simulation execution. HELICS provides flexible time management mechanisms but requires explicit configuration of logical time to synchronise each federate. Different model combinations require different synchronisation approaches based on their coupling strength, characteristic timescales, and numerical properties. Models in the catalog specify an *sdm:OutputTimeInterval* representing their internal simulation time advancement. However, logical time coordination properties are defined at the topology level rather than within individual model specifications, enabling synchronisation parameters to be configured dynamically based on the coupling context. This topology-level specification is implemented through HELICS period and offset parameters.

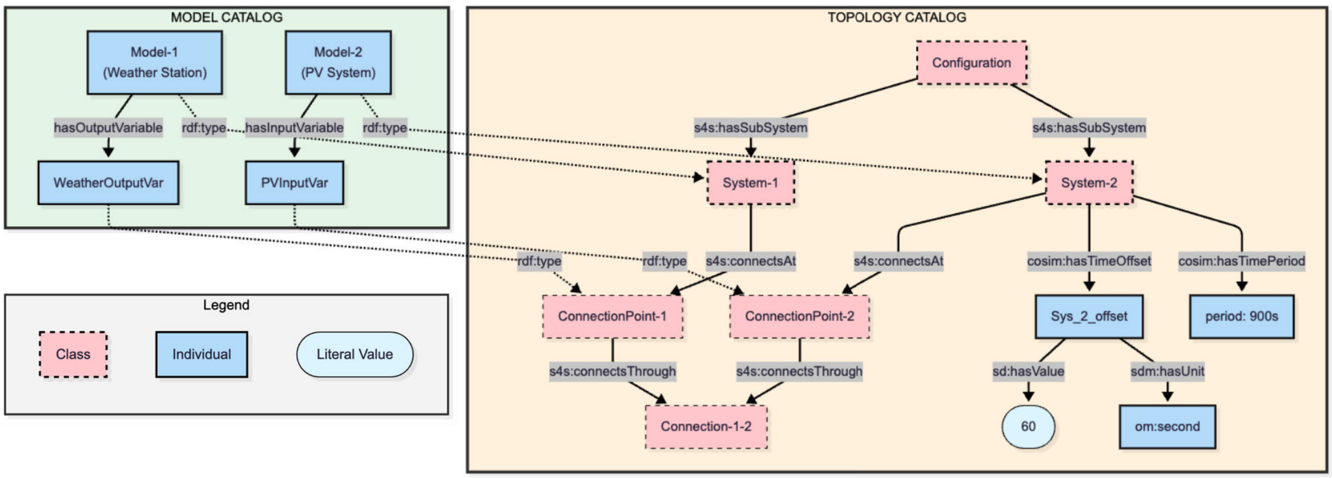

FIGURE 2. **Semantic modelling demonstration using graph-based representation of two Models catalogue entries connected via Topologies catalog patterns, including subsystem definitions, connection points, and variable mappings.**

```
PREFIX sdm: <http://w3id.org/sdm#>
PREFIX sd: <http://w3id.org/okn/o/sd#>
PREFIX om: <http://www.ontology-of-units-of
    -measure.org/resource/om-2/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX catalog: <http://model_catalog#>

ASK WHERE {
  BIND(catalog:model1 AS ?model1)
  BIND(catalog:model2 AS ?model2)
  BIND(catalog:variable1 AS ?var1)
  BIND(catalog:variable2 AS ?var2)
  {
    ?model1 sdm:hasOutputVariable ?var1 .
    ?model2 sdm:hasInputVariable ?var2 .
  } UNION {
    ?model1 sdm:hasInputVariable ?var1 .
    ?model2 sdm:hasOutputVariable ?var2 .
  }
  ?var1 sdm:hasStandardVariable ?standardVar .
  ?var2 sdm:hasStandardVariable ?standardVar .
  ?var1 sdm:hasDataType ?datatype .
  ?var2 sdm:hasDataType ?datatype .
  ?var1 sdm:usesUnit ?unit1 .
  ?var2 sdm:usesUnit ?unit2 .
  ?unit1 om:hasDimension ?dimension .
  ?unit2 om:hasDimension ?dimension .
  ?var1 sdm:hasMinValue ?min1 .
  ?var1 sdm:hasMaxValue ?max1 .
  ?var2 sdm:hasMinValue ?min2 .
  ?var2 sdm:hasMaxValue ?max2 .
  FILTER(?max1 >= ?min2 && ?max2 >= ?min1)
}
```

Listing 1. SPARQL query for variable compatibility verification.

Within each topology, system classes are annotated with *cosim:hasTimePeriod* and *cosim:hasTimeOffset* properties. The period parameter defines the logical time interval at which a federate advances, while the offset parameter delays federate execution with respect to the period to ensure alignment with data availability from upstream models.

When models are instantiated within a specific topology, they inherit these context-appropriate synchronisation parameters. This approach provides substantial reusability: timing configurations defined once within topology templates can be applied across multiple scenario instances, reducing the configuration burden on domain experts. While HELICS offers additional sophisticated timing features, including various synchronisation modes and fine-grained control mechanisms, the period and offset parameters employed here prove sufficient for the coupling patterns demonstrated in this work for all feasible model combinations.

E. THE DATA-GRAPH

While the Model Catalog formalises simulation components and the Topologies Catalog defines their interconnections across different contexts, the Data Graph provides the semantic representation of the physical systems being simulated. Building on the ontologies reviewed in the previous section (SAREF, BEM, GeoSPARQL), the Data Graph leverages semantic technologies to integrate heterogeneous urban data sources into a unified representation. Urban energy modelling has become essential for managing energy use within cities, with numerous methodologies and tools developed to assess energy demands, benchmark performance, and analyse consumption patterns [25]. However, a fundamental requirement is to enable domain experts to declaratively specify simulation scenarios without requiring in-depth knowledge of simulation models and their technical requirements. The Data Graph takes advantage of ontological modelling to enable the framework to understand declarations and to create the underlying co-simulation infrastructure. An overview of the ontology concepts used and their relations is shown in Figure 3. Although the majority of concepts necessary for the demonstration were already established in existing ontologies, the *new:RoofPart* concept was introduced to represent individual roof surfaces and their associated geometric and physical attributes.

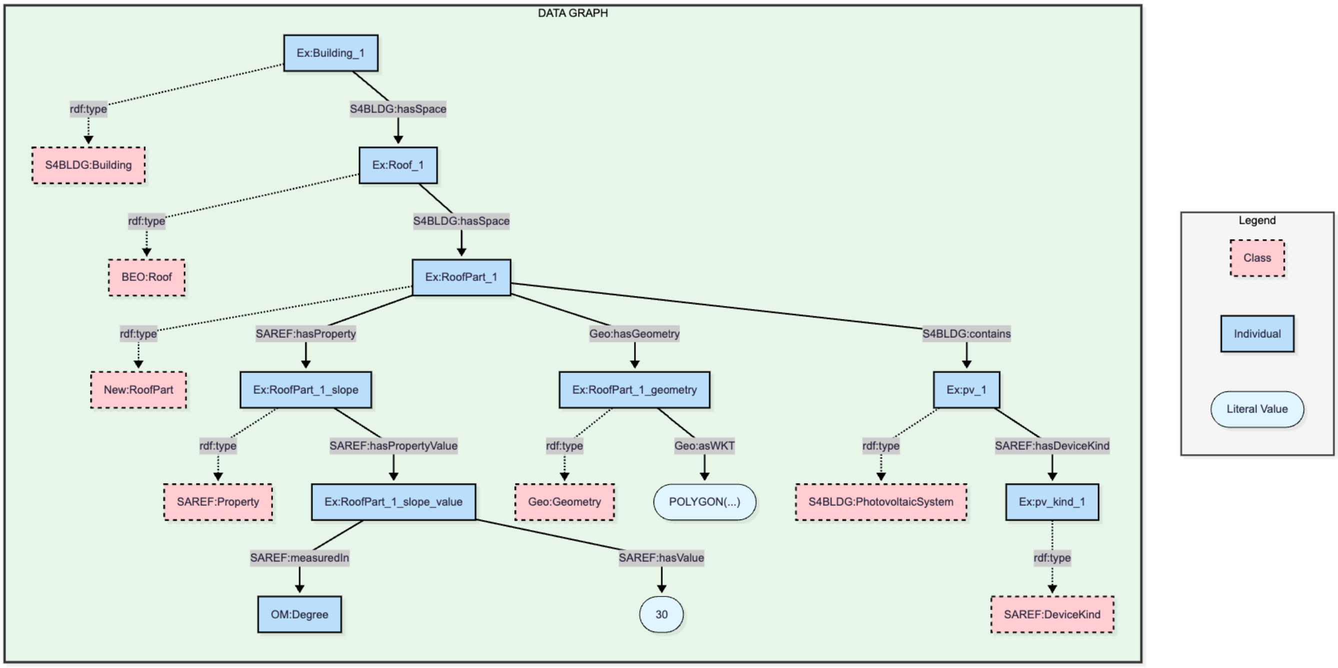

FIGURE 3. Example instantiation demonstrating the ontological framework for representing a building structure with roof surfaces and an associated photovoltaic system.

This section illustrates the methodology through an example use case of a photovoltaic installation, demonstrating how building structures and rooftop properties are represented. In this scenario, a domain expert specifies that "Rooftop_1 features a pitched surface with a 30° inclination that hosts a PV panel from a specified manufacturer" without requiring knowledge of which simulation model will represent this system or what input parameters that model demands. The example in Listing 2 illustrates how ontologies enable the creation of declarative system specifications that can be automatically processed by the framework.

The declarative statements were formulated using Protégé [27], a widely-adopted ontology development environment. While manual creation is possible, these statements are typically derived from existing data sources.

As discussed in the introduction, urban data are dispersed across multiple sources and formats, necessitating a unified representation. To this end, RML mappings provide a declarative framework for transforming data from various source formats into RDF triples that conform to the ontological schema. The execution of these mappings requires a graph materialisation engine capable of efficiently processing heterogeneous data sources and executing the transformation rules at scale. For this purpose, we employ Morph-KGC [1], selected for its performance characteristics in constructing knowledge graphs. Although any data format could be used for demonstration purposes, this work leverages a CityGML 2.0 dataset hosted in a 3DCityDB [22] instance, from which building models and rooftop geometries are extracted and materialized. In addition, RDF flexibility enables knowledge enrichment at multiple levels. First, we integrate technical specifications by leveraging *saref:DeviceKind* concept to incorporate manufacturer datasheet parameters for PV panels. This enrichment extends the representation beyond purely geometric attributes to include essential technical parameters required for simulation configuration. Second, the framework enables derivation of implicit properties from existing data through computational enrichment. Properties that are not directly available in source datasets but can be computed from existing information are systematically derived and materialized within the knowledge graph. For instance, roof slope values, implicit in CityGML 2.0 geometries but critical for calculating PV energy production, are derived from geometric representations via external processes, with results materialized in the graph through SPARQL queries. This pattern decouples domain-specific computations from the ontology structure while ensuring derived knowledge remains queryable, enabling direct retrieval by downstream configuration services. Importantly, this computational enrichment pattern can extend beyond geometric properties to encompass any derived knowledge: energy performance indicators, system compatibility assessments, or optimization parameters. This showcases the framework's extensibility across diverse domain requirements. Although our implementation focuses on a specific use case, comprehensive transformation of CityGML 2.0 data into a knowledge graph falls outside the scope of this study; rather, this work establishes foundational principles for ontology-driven automation that can be extended to broader simulation scenarios and diverse modelling tools. Combined with the Model and Topologies

```
@prefix s4bldg: <https://saref.etsi.org/saref4bldg/> .
@prefix beo: <https://pi.pauwel.be/voc/buildingelement>
     .
@prefix rdf:
     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix saref: <https://saref.etsi.org/core/> .
@prefix om: <http://www.ontology-of-units-of
-measure.org/resource/om-2/> .
@prefix ex: <http://example-scenario-data#> .

ex:Building_1
    rdf:type s4bldg:Building ;
    s4bldg:hasSpace ex:Rooftop_1 .
ex:Rooftop_1
    rdf:type beo:Roof ;
    s4bldg:hasSpace ex:RoofPart_1 .
ex:RoofPart_1
    rdf:type new:RoofPart ;
    saref:contains ex:PhotovoltaicSystem_1 ;
    saref:hasProperty ex:Slope_1 .
ex:Slope_1
    rdf:type saref:Property ;
    saref:hasPropertyValue ex:Slope_1_Value .
ex:PhotovoltaicSystem_1
    rdf:type s4bldg:PhotovoltaicSystem ;
    saref:hasDeviceKind ex:pv_series_M .
value:Slope_1_Value
    rdf:type saref:PropertyValue ;
    saref:isMeasuredIn om:Degree ;
    saref:hasValue "30"^^xsd:double .
```

Listing 2. Example declaration of a photovoltaic system, illustrating the integration of existing data (non-bold) with domain expert modifications (bold).

Catalog, the Data Graph complements the semantic foundation for automated co-simulation construction.

F. SPARQL-BASED QUERYING SYSTEM FOR CO-SIMULATION COMPOSITION

Given the Data-Graph specifying buildings and their equipment, the Model Catalog describing available simulation models, and the Topologies Catalog defining coupling patterns, we execute the composition workflow presented in Algorithm 1 to generate a Co-simulation Graph that synthesizes the complete co-simulation scenario (see Querying System in Figure 1). By leveraging concept reuse between the Data Graph and Model Catalog, the algorithm first identifies all buildings and their installed equipment within the Data Graph. For each building, it searches for a topology configuration capable of simulating the required system composition. Listing 3 shows an example query used to identify such configuration topologies. For each system in the identified configuration, the algorithm retrieves the necessary metadata from the Model Catalog and creates a model instance in the Co-simulation Graph, incorporating execution information, interface variables, and their corresponding connection point types from the Topologies Graph.

Model instances representing the systems within a building are grouped under a building-specific node and linked to the corresponding data graph entity via the *cosim:belongsTo* property. For example, the PV system installed on roof pitch 8 of building 51 is linked directly to that roof pitch entity through this relationship, establishing explicit traceability between simulation components and physical building elements.

Algorithm 1 Co-simulation composition workflow

```
Initialize storage, knowledge manager, and scenario graph
Create storage bucket
buildings ← Query buildings from data-graph
for all building do
    required_systems ← Identify building systems
    configuration ← Find matching configuration for required_systems in topology-graph
    for all model in configuration do
        Retrieve metadata from models-graph
        if model requires initialization then
            Run initialization service → Store executable file in bucket
        end if
        Add model instance to co-simulation graph with:
            - Metadata (name, type, execution command, executable file)
            - Temporal parameters (time intervals)
            - Input/output variables (with connection types)
    end for
end for
for all input variable in co-simulation graph do
    compatible_outputs ← Find outputs with matching connection types in cosimulation graph
    Create directed connections: output → input
end for
Serialize scenario graph to file
```

When a model requires initialization, the algorithm invokes its associated generation service, passing the building reference to enable retrieval of contextual information for generating model-specific configuration files. The resulting file is semantically linked to the model instance in the Co-simulation Graph via the *sd:hasInput* property and persisted in object storage for subsequent delivery to the corresponding federate upon instantiation. Subsequently, the algorithm establishes variable connections by querying the topology graph for matching connection types and creating directed edges from compatible outputs to inputs using the *cosim:directedTo* property. Listing 4 shows an excerpt of the final co-simulation graph document generated at this stage, ready to be translated into an HELICS federation.

At this stage, all executable files are stored in object storage (a storage system for unstructured data and binary files) and linked to their corresponding model instances in the Co-simulation Graph. This graph contains the complete federation specification: all model instances, their interconnections, and synchronization parameters. The system uses this information to regenerate model interfaces with the right connectivity and timing, and to deliver the correct

```
PREFIX s4s: <http://w3id.org/s4s#>
PREFIX sdm: <http://w3id.org/sdm#>
PREFIX rdf:
     <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX s4bldg: <https://saref.etsi.org/saref4bldg/> .
SELECT DISTINCT ?conf ?modelconf0 ?modelconf1
WHERE {
  ?conf s4s:hasSubSystem ?subsystem0 .
  ?modelconf0 rdf:type sdm:ModelConfiguration .
  ?modelconf0 rdf:type ?subsystem0 .
  ?modelconf0 rdf:type s4bldg:PhotovoltaicSystem .

  ?conf s4s:hasSubSystem ?subsystem1 .
  ?modelconf1 rdf:type sdm:ModelConfiguration .
  ?modelconf1 rdf:type ?subsystem1 .
  ?modelconf1 rdf:type s4bldg:WeatherStation .

  ?conf s4s:hasSubSystem ?anySubsystem .
}
GROUP BY ?conf ?modelconf0 ?modelconf1
HAVING (COUNT(DISTINCT ?anySubsystem) = 2)
```

Listing 3. SPARQL query against the topology graph for finding topology configurations matching specified building equipment. Example limited to two-element topologies comprising a PV system and a weather station.

configuration files to each federate when it starts up. The final workflow is shown in Figure 4.

IV. SCENARIO

A. USE CASE OVERVIEW

To demonstrate COSIMO's capabilities across varying spatial scales and levels of distributed energy resource penetration, we implement a comparative study of three urban configurations within the city of Aachen, comprising 50, 100, and 500 residential buildings respectively, each evaluated under four distinct technology deployment scenarios. The city of Aachen was selected for this analysis due to the public availability of CityGML 2 Level of detail (LoD)2 datasets [39], which provide detailed three-dimensional building geometries and semantic information essential for urban energy modeling. This approach reflects realistic planning contexts where urban energy planners evaluate progressive integration of renewable technologies and storage systems across neighborhoods of different sizes. All scenarios simulate one year of operation with hourly timesteps to capture representative energy flow patterns. While adding equipment to individual buildings is feasible through Protégé's graphical interface, we employed SPARQL to programmatically edit blocks of buildings and assign technologies at scale, enabling efficient scenario configuration across all districts.

All scenarios include meteorological monitoring, occupancy-driven load profiles, and smart metering infrastructure as baseline components that will be discussed in the next paragraph. Scenario A represents the baseline condition where buildings are equipped only with monitoring and consumption modeling capabilities, representing the current state without distributed power generation. Scenario B introduces renewable energy sources with low penetration with approximately one quarter of buildings equipped with rooftop PV systems. Scenario C doubles this to medium penetration with half the buildings featuring PV installations. Finally, Scenario D maintains the same PV penetration level but adds battery energy storage systems to capture locally generated power. Each scenario exercises different topology patterns, demonstrating the framework's ability to automatically select and instantiate appropriate coupling configurations based on building equipment specifications across different urban scales.

B. BUILDING CHARACTERISTICS

Building geometry and spatial data originate from CityGML 2.0 datasets at LoD 2 for each of the three districts. During the knowledge graph enrichment process, roof geometric properties are derived and added to the data graph for all buildings, including surface altitude, orientations, slope, and area.

For photovoltaic installations, suitable roof surfaces are defined as those with azimuth angles between 135° and 225°, corresponding to south-facing orientations optimal for solar energy capture. The PV model generation service computes the installed capacity by considering panel dimensions, rooftop surface area, and assuming 75% coverage of the available surface to account for spacing and mounting constraints.

C. SIMULATION MODELS

The co-simulation framework employs a modular architecture where individual building energy systems are represented through interconnected component models, starting from a baseline weather-meter configuration. The topologies catalog includes pre-validated topologies for each combination of photovoltaic systems, battery storage, and occupancy models that can be integrated with the baseline configuration. This combinatorial approach ensures that any technically valid system configuration encountered in the scenario specifications can be matched to a corresponding topology.

Weather conditions are provided through a CSV file reader that retrieves temporal weather data from the OpenWeather APIs [29]. The model operates as a data source federate, outputting three key meteorological variables: global horizontal irradiance (GHI) in W/m^2, external air temperature (T_{ext}) in °C, and relative humidity (RH) in percentage.

Solar electricity generation is simulated using a PV model based on Bottaccioli et al. [4]. The model accepts GHI (W/m^2) and external temperature (°C) as inputs and computes the DC power output (W). The model accounts for temperature-dependent efficiency losses and irradiance-to-power conversion based on panel characteristics. Panel specifications come from manufacturer datasheets aggregated in JSON format. The data encompasses technical parameters for a single PV panel model distributed across installations in all three district scales, as shown in Table 2. This module represents a typical commercial panel with a nominal power of 435 watts and efficiency of 21.2%.

Residential electricity demand profiles are generated using a stochastic occupancy-based simulator implementing the Richardson method [34]. This model generates

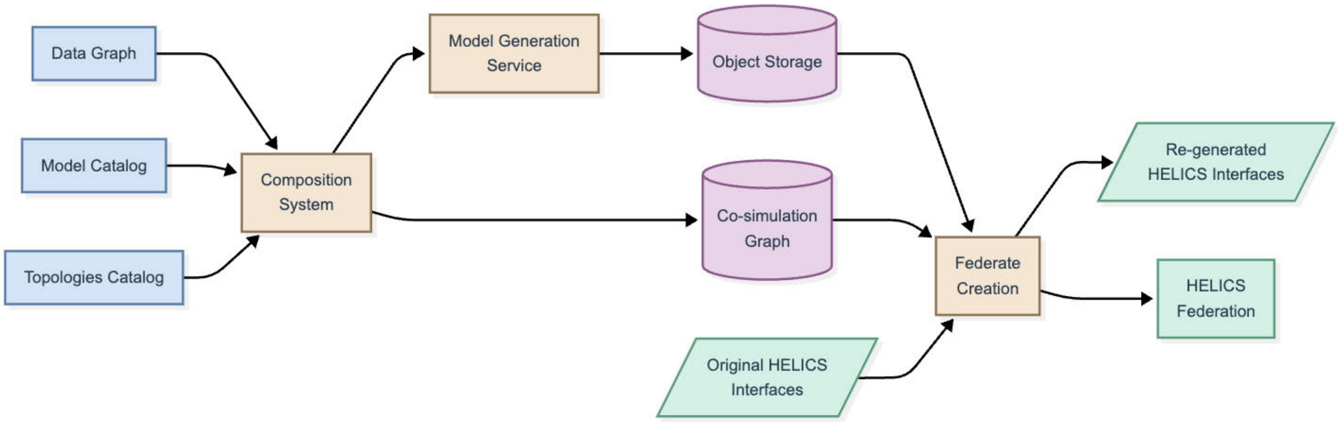

FIGURE 4. Co-simulation federation generation pipeline showing the transformation from semantic graphs through composition querying, model generation, and federate creation to produce executable HELICS federations.

TABLE 2. Photovoltaic module specifications.

Parameter	Value
Nominal Power ($P_{\max}$)	435 W
Efficiency (η)	21.2%
Temperature Coefficient ($P_{\max}$)	-0.34%/°C
Module Dimensions	2.094×1.038 m

realistic household electrical load profiles by simulating occupant behavior and appliance usage patterns. Operating autonomously without external inputs, the model outputs electrical power consumption (P_{usage}) in watts, capturing the temporal variability characteristic of residential energy demand.

Energy storage is represented through a Modelica-based Functional Mock-up Unit (FMU) implementing a simplified battery model. The model operates bidirectionally, absorbing power (positive flow) during charging periods and supplying power (negative flow) during discharge cycles. A single lithium-ion battery technology type is deployed in Scenario D across all district scales, with specifications detailed in Table 3. The battery model was developed in Modelica and exported as an FMU for co-simulation integration, representing a typical residential storage system suitable for coupling with photovoltaic installations.

The electrical power balance at each building node is computed by a Python-based smart meter model that functions as an aggregator. The meter accepts power production inputs (P_{in}) from distributed generation sources and power consumption inputs ($\text{power}_{\text{req}}$) from building loads, both measured in watts. The model calculates and outputs the net power flow (P_{net} in watts), representing the algebraic sum of all connected power sources and sinks. This net power value indicates whether the building is drawing from or supplying to the grid at each simulation interval.

Based on what is defined in the data graph, the composition algorithm (see Algorithm 1) identifies and instantiates an appropriate topology for each building across all district scales. Table 4 presents the distribution of topologies across scenarios and district scales: Scenario A establishes the baseline with only weather conditions, occupancy models and meters. Scenario B introduces photovoltaic systems to 26% of buildings, Scenario C implements 50% PV penetration, and Scenario D pairs PV with battery storage on 50% of buildings using the complete five-component topology.

The *instances* count represents the total number of model federates requiring instantiation and orchestration, while *connections* indicate the data exchange links demanding runtime coordination. System complexity scales significantly from 150 instances and 100 connections in the 50-building baseline to 2000 instances and up to 1,260 connections in the 500-building scenarios. Manual configuration of such large-scale simulations would be prohibitively labor-intensive and error-prone, particularly given that each connection must be correctly specified in multiple locations to ensure bidirectional consistency between federate interfaces. A single misconfigured connection, whether due to incorrect naming, incompatible units of measurement, or inconsistent data type specifications, can compromise the entire simulation ensemble. The framework's automated composition and orchestration capabilities directly address this challenge, eliminating human error in connection mapping. Once topologies are selected, the framework generates technology-specific configuration files for each model instance, accounting for the varied initialization requirements and interface specifications inherent to different simulation tools.

D. GENERATED CONFIGURATION ARTIFACTS

Model initialization requirements vary according to technology characteristics, with dedicated services handling contextual parameterization where necessary.

Weather models leverage an initialization service that queries the OpenWeather API based on building coordinates, retrieving location-specific meteorological time series data with hourly granularity. Photovoltaic model initialization employs the model generation service to produce JSON

```
@prefix s4bldg: <https://saref.etsi.org/saref4bldg/> .
@prefix beo: <https://pi.pauwel.be/voc/buildingelement>
     .
@prefix rdf:
     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix saref: <https://saref.etsi.org/core/> .
@prefix om:
     <http://www.ontology-of-units-of-measure.org/resource
     .
@prefix sd: <https://w3id.org/okn/o/sd#> .
@prefix sdm: <http://w3id.org/sdm#> .
@prefix schema: <http://schema.org/> .
@prefix s4s: <https://saref.etsi.org/saref4syst/> .
@prefix cosim: <http://example-cosimulation#> .
@prefix catalog: <http://example-catalog#> .
@prefix topo: <http://example-topologies#> .
@prefix data: <http://example-data#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

cosim:pv_bldg_51 rdf:type s4s:System ;
    rdf:type topo:pv_eval_sys_pv ;
    schema:name "pv_sim_federate.py" ;
    sd:hasExecutionCommand "python" ;
    sdm:hasInput cosim:pv_bldg_51_input_file ;
    sdm:hasOutputTimeInterval
     cosim:pv_bldg_51_time_interval ;
    topo:hasTimeOffset cosim:pv_bldg_51_offset ;
    topo:hasTimePeriod cosim:pv_bldg_51_period ;
    cosim:belongsTo data:Building_51 .

cosim:meter_bldg_51 rdf:type s4s:System ;
    schema:name "meter_federate.py" ;
    sd:hasExecutionCommand "python" ;
    sdm:hasOutputTimeInterval
     cosim:meter_bldg_51_time_interval ;
    cosim:belongsTo data:bldg_51_roof_8 .

cosim:pv_bldg_51_input_file rdfs:label
     "roofpitch_8_pvsys.json" .
cosim:meter_bldg_51_time_interval sdm:hasIntervalValue
     "3600"^^xsd:integer .
cosim:meter_bldg_51_offset topo:hasOffsetValue
     "0.2"^^xsd:decimal .
cosim:meter_bldg_51_period topo:hasPeriodValue
     "1"^^xsd:integer .
cosim:pv_bldg_51_time_interval sdm:hasIntervalValue
     "3600"^^xsd:integer .
cosim:pv_bldg_51_time_interval sdm:usesUnit om:second .
cosim:pv_bldg_51_offset topo:hasOffsetValue
     "0.1"^^xsd:decimal .
cosim:pv_bldg_51_period topo:hasPeriodValue
     "1"^^xsd:integer .
cosim:pv_bldg_51 sdm:hasOutputVariable
     cosim:pv_out_power_51 .

cosim:pv_out_power_51 rdf:type topo:connp_output_power ;
    sd:hasShortName "power_dc" .

cosim:meter_bldg_51 sdm:hasInputVariable
     cosim:meter_in_power_51 .

cosim:meter_in_power_51 rdf:type topo:connp_input_power
     ;
    sd:hasShortName "P_in" .

cosim:pv_out_power_51 catalog:directedTo
     cosim:meter_in_power_51 .
```

Listing 4. Excerpt from the co-simulation graph depicting two model instances with their associated variables, inter-model connection, and configuration file dependencies.

configuration files encoding roof geometric properties such as slope and azimuth, along with manufacturer specifications corresponding to the deployed panel model.

In contrast, occupancy behavior is governed by stochastic processes executed during simulation runtime, requiring no prior initialization. Similarly, smart meter models operate without initialization requirements, directly monitoring building consumption. Battery systems utilize standardized technology parameters independent of building-specific attributes, thus eliminating the need for contextual initialization data.

TABLE 3. Battery energy storage system specifications.

Parameter	Value
Total Energy Capacity	14.0 kWh
Operating Temperature Range	−20°C to +50°C
State of Charge (SoC) Range	5–100%

TABLE 4. Topology Pattern Distribution Across Scenarios and District Scales.

Topology	Scenario			
	A	B	C	D
50 Buildings				
W + O + M	50	37	25	25
W + O + M + PV	0	13	25	0
W + O + M + PV + B	0	0	0	25
Instances	150	185	200	200
Connections	100	126	125	125
100 Buildings				
W + O + M	100	74	50	50
W + O + M + PV	0	26	50	0
W + O + M + PV + B	0	0	0	50
Instances	300	370	400	400
Connections	200	252	250	250
500 Buildings				
W + O + M	500	370	250	250
W + O + M + PV	0	130	250	0
W + O + M + PV + B	0	0	0	250
Instances	1500	1850	2000	2000
Connections	1000	1260	1250	1250

W: Weather; O: Occupancy; PV: Photovoltaic; B: Battery; M: Meter.

Having established four distinct scenarios ranging from baseline monitoring to fully integrated PV-battery systems across three district scales, the automated composition workflow has generated complete, executable co-simulation environments for each configuration. The framework successfully handled scenarios ranging from 150 model instances in the smallest scenario to 2000 instances in the largest, while managing the requisite initialization services and coupling specifications. What would have required hours to weeks of manual configuration and error-prone coupling definitions was accomplished programmatically through semantic reasoning and automated instantiation.

The next section examines the simulation outputs to validate the correctness of this automated process and demonstrate the framework's practical value for rapid urban energy planning studies across different urban scales.

V. EXPERIMENTAL RESULTS

This section evaluates COSIMO's automated co-simulation construction workflow across multiple district scales with increasing renewable energy penetration. We conducted experiments at three urban scales comprising 50, 100, and 500 buildings respectively, each evaluated under four technology deployment scenarios. We assess both the automation efficiency through time savings and computational performance, and the physical correctness of generated simulation configurations through energy performance metrics. All experiments were carried out on a consumer-grade laptop (Intel i7-8750H, 2.20 GHz, 16 GB RAM), used exclusively for the co-simulation synthesis stage, while co-simulation execution was performed on a distributed infrastructure. Resource usage was measured by averaging samples collected every 250 ms over ten repeated runs. The pipeline comprises three sequential phases: data materialization, composition engine, and model generation service. Since data materialization performance is extensively discussed in the Morph-KGC literature, the analysis focuses on the remaining two phases. At $n = 10$, both phases remain within moderate resource limits: the composition engine peaks at approximately 760 MB RAM and 18% CPU, while the model generation service reaches 1.2 GB RAM and 56% CPU. Scaling up to $n = 500$, the composition engine remains manageable at 2.7 GB RAM and 41% CPU, whereas the model generation service becomes the dominant cost component, consuming up to 14.2 GB RAM at 87% average CPU. This is primarily due to the concurrent spawning of multiple Docker containers, whose aggregate load explains why memory capacity becomes the main bottleneck at larger scales rather than the semantic composition logic itself. The architecture is nevertheless naturally suited to horizontal scaling, since each container processes an independent entity batch, distributing the workload across multiple machines would proportionally reduce memory pressure and shift the scalability limit from a single workstation to the resources of a distributed computer network.

To evaluate the physical correctness and energy performance of the automatically generated co-simulations, we first define the key performance indicators used throughout this analysis. We track four primary energy flows: total building consumption (E_{cons}), which represents the annual aggregate electrical energy consumed by all buildings in the scenario; total PV generation (E_{PV}), the annual aggregate electrical energy generated by all photovoltaic systems; grid import (E_{import}), the annual energy imported from the grid to meet building demand; and grid export (E_{export}), the annual energy exported to the grid when local generation exceeds local consumption.

From these basic energy flows, we derive three performance indicators. The self-consumption rate (SC) represents the fraction of locally generated PV energy that is consumed locally rather than exported to the grid:

$$SC = \frac{E_{\text{PV}} - E_{\text{export}}}{E_{\text{PV}}} \times 100\% \tag{1}$$

The self-sufficiency rate (SS) measures the fraction of total building consumption met by local PV generation:

$$SS = \frac{E_{\text{PV}} - E_{\text{export}}}{E_{\text{cons}}} \times 100\% \tag{2}$$

Finally, the grid dependency reduction (GDR) quantifies the percentage reduction in grid imports compared to the baseline scenario without distributed generation:

$$GDR = \frac{E_{\text{import}}^{\text{baseline}} - E_{\text{import}}}{E_{\text{import}}^{\text{baseline}}} \times 100\% \tag{3}$$

All performance indicators represent annual aggregates calculated as the ratio of cumulative energy flows over the entire simulation period (8760 hours i.e. one year).

A. PHYSICAL VALIDATION OF SIMULATION OUTPUTS

Beyond demonstrating automation efficiency, we verify that automatically generated configurations produce physically correct and meaningful results through energy performance analysis across all scenarios. This validation approach is consistent with established co-simulation methodologies for energy communities [5], where scenario-based energy performance analysis serves as a means to assess the physical correctness and practical relevance of simulation outputs. The objective here is therefore not a formal numerical stability analysis of the runtime co-simulation process, but rather an assessment of whether the automatically generated configurations yield physically plausible and internally consistent energy results appropriate for the scenario-construction focus of the proposed framework. We analyze annual energy metrics computed from hourly simulation data across all scenarios and district scales to demonstrate the progressive impact of PV and battery deployment. Figure 5 illustrates the daily energy flows for a representative 50-building district across the four analyzed scenarios, visualizing how different distributed energy resource configurations affect the balance between local generation, storage, and grid interaction throughout a typical day.

Total building consumption remains proportional to district size, maintaining constant per-building consumption of approximately 17 MWh annually across all scenarios. The load profile (black dashed line in Figure 5) exhibits typical residential patterns with morning activity peaks around 7-8 AM and pronounced evening peaks near 7 PM, followed by reduced nighttime consumption.

Scenario A establishes the baseline condition without distributed generation, showing complete grid dependency with all consumption met through grid imports (100% grid dependency), represented by the blue area perfectly matching the load profile throughout the day.

Scenario B introduces photovoltaic systems to 25% of buildings, generating approximately 7.9 MWh per system annually. As shown in Figure 5, PV generation occurs exclusively during daylight hours, with peak production around midday. The green hatched area represents direct self-consumption of solar energy, while the pink area below the axis indicates excess generation exported to the grid during peak production hours. The simulation results show that 60% of this generated energy is consumed locally, while the remaining 40% is exported to the grid. Grid imports remain necessary during morning, evening, and nighttime hours when solar generation is insufficient or absent. This configuration achieves a 7% reduction in aggregate grid imports compared to baseline, demonstrating measurable impact from limited PV deployment.

Scenario C doubles PV penetration to 50% of buildings, maintaining the same per-system generation capacity. The increased deployment improves annual self-consumption to 65%, as higher local generation better aligns with aggregate consumption patterns. The visualization reveals substantially larger self-consumption (green hatched area) during daylight hours and correspondingly greater grid exports (pink area) during midday peak production. Grid imports decrease by 15% relative to baseline, though absolute grid export increases proportionally with the larger installed capacity. This scenario reveals the diminishing returns of self-consumption improvement with increasing penetration in the absence of energy storage, as evidenced by the growing mismatch between the timing of peak solar generation and consumption patterns.

Battery integration in Scenario D produces the most significant impact on energy flows. While maintaining the same 50% PV penetration as Scenario C, the addition of battery storage systems with 14 kWh capacity enables 85% annual self-consumption, representing a 20% point improvement. Figure 5 clearly illustrates the role of batteries through two distinct patterns: the purple hatched area below the axis represents battery charging during periods of excess solar generation (primarily midday hours), while the gray hatched area above the axis shows battery discharge during consumption peaks (evening and night hours). Battery systems store excess solar generation during high-production periods and discharge during consumption peaks, substantially improving the alignment between generation and demand. The plot demonstrates how storage enables temporal shifting of solar energy, reducing both grid imports during evening hours and grid exports during midday. This configuration achieves a 20% reduction in grid imports compared to baseline and a 57% reduction in grid export compared to Scenario C, demonstrating the critical role of storage in maximizing local energy utilization.

The consistency of percentage-based metrics across all three district scales validates the correct instantiation and coupling of simulation models. Self-consumption rates of 60%, 65%, and 85% for Scenarios B, C, and D respectively remain almost constant across 50, 100, and 500 building districts, confirming that the automated composition process correctly replicates topologies at different scales. Self-sufficiency levels reach 7%, 15%, and 20% for the three PV scenarios, values that accurately reflect the partial deployment of generation assets where only 25% or 50% of buildings contribute local generation.

Grid dependency reductions correspond appropriately with both technology penetration and district size, demonstrating that the framework maintains physical consistency as complexity increases. The automated coupling between PV generation models, battery dynamics, and occupancy consumption profiles produces energy flows that match expected patterns for residential districts with distributed energy resources. These results collectively indicate that COSIMO's automated approach maintains physical correctness and produces reliable simulation outputs across scales ranging from neighborhood to large urban district levels.

B. WORKFLOW AUTOMATION BENEFITS

The primary contribution of COSIMO lies in automating the co-simulation construction workflow. Table 5 compares the traditional manual approach against COSIMO's ontology-driven automation across the three district scales for scenarios A and D, which represent the boundary cases.

TABLE 5. Workflow time comparison across district scales and scenarios.

	Number of Buildings		
Task	**50**	**100**	**500**
Manual Approach - Scenario A			
Model instantiation	3 hours	6 hours	30 hours
Configuration files	4 hours	8 hours	40 hours
Parameter mapping	2 hours	4 hours	20 hours
Total expert time	∼15 h	∼30 h	∼150 h
Manual Approach - Scenario D			
Model instantiation	4 hours	8 hours	40 hours
Configuration files	6 hours	12 hours	60 hours
Parameter mapping	2 hours	4 hours	20 hours
Total expert time	∼20 h	∼40 h	∼200 h
COSIMO Automated Approach			
Composition (Scen. A)	23 s	42 s	186 s
Composition (Scen. D)	56 s	105 s	524 s
Time reduction	99.95%	99.95%	99.95%

Manual time estimates are based on our experience with urban energy co-simulation projects and represent an approximate breakdown of expert effort required for configuration tasks. These estimates assume practitioners are familiar with the simulation tools but must manually specify connections, instantiate models, and generate configuration files. For the 500-building district in Scenario D, manual configuration would require approximately 200 hours of expert time, equivalent to five weeks of full-time work. COSIMO's automated approach reduces practitioner effort to initiating the composition process, with actual configuration generation completing in minutes even for the largest and most complex scenario. This assumes pre-existing data mappings, an annotated model catalog, validated topologies, and available model generation services. While these preparatory activities are time-consuming, they constitute one-time investments reused across all future scenarios.

The dominant phase across all scales is model generation service invocation, accounting for 35% to 45% of total time.

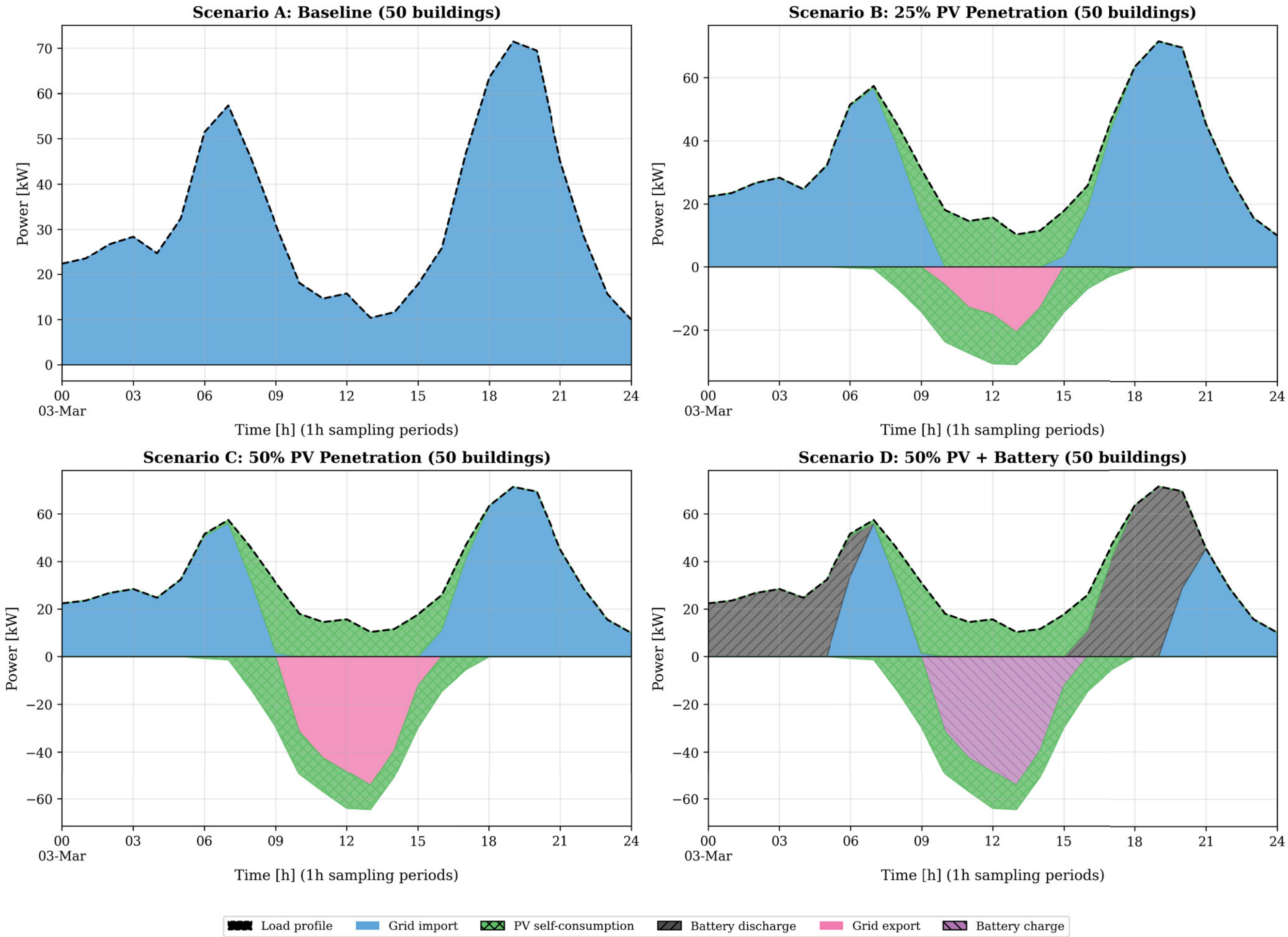

FIGURE 5. Daily energy flow patterns for a 50-building district demonstrating progressive impact of distributed energy resources during 24 hours.

This phase offers optimization opportunities through parallelization in future implementations. For the 500-building district in Scenario D, which requires instantiation and connection of 2000 model instances, composition completes in approximately nine minutes. The time savings relative to manual approaches remain consistently above 99.9% across all tested scales, demonstrating that automation benefits are preserved at larger scales where manual configuration becomes increasingly impractical.

VI. DISCUSSION AND LIMITATIONS

This work demonstrates that ontology-driven approaches can successfully automate co-simulation construction for urban multi-energy systems, achieving dramatic time savings while maintaining physical correctness. The framework addresses critical gaps in existing approaches by integrating semantic representation, partial automated validation, and systematic parameter mapping into a unified workflow. Results confirm that semantic web technologies are sufficiently mature to support practical urban energy modeling applications at scale.

Several limitations indicate directions for future research. The adoption of a unified semantic schema, while central to COSIMO's automation, introduces its own limitations. Updates to upstream standards such as SAREF propagate across all dependent mappings, requiring coordinated maintenance effort that grows with graph size. The approach also entails considerable upfront modelling effort, as data sources, model interfaces, and coupling patterns must all be aligned with the shared ontology before automation benefits materialize. Heterogeneity is furthermore not fully removed but often displaced into mappings and adapters, which remain costly to maintain and difficult to debug. The shared schema also forces heterogeneous systems into a common representation that may not naturally fit all of them, flattening tool-specific assumptions and omitting fine-grained distinctions. Finally, semantic ambiguity represents a concrete risk when the same concept carries different meanings across domains, a challenge particularly pronounced during model annotation phase, where different modelers may interpret shared vocabulary inconsistently, compromising the reliability of automated reasoning across the graph.

Model generation services and topology patterns require initial manual definition, representing upfront investment before automation benefits materialize. However, once established, these definitions become reusable assets.

The current topology validation framework ensures syntactic compatibility and physical meaningfulness of component connections. However, it does not yet incorporate comprehensive system-level design constraints. Specifically, the validation does not include numerical stability analysis, detection of self-loops, or identification of potential deadlock configurations. These advanced validation capabilities represent important directions for future work and will enhance the robustness of the topology verification process.

The implementation couples tightly with HELICS, and supporting alternative co-simulation frameworks would require additional abstraction layers. Multi-carrier energy systems extending to thermal, gas, or hydrogen networks require ontology extensions, though the modular architecture supports such expansions.

Besides these limitations, this work represents a significant step toward the automatic creation of digital copies for urban energy systems. While the current implementation generates static co-simulation configurations from available data, the semantic infrastructure developed here lays crucial groundwork for evolving toward digital twins with real-time data integration, continuous model calibration, and operational feedback capabilities.

VII. CONCLUSION

This work presented COSIMO, an ontology-driven framework that addresses the challenge of constructing co-simulations for urban multi-energy systems at scale. Through systematic integration of semantic web technologies, the framework successfully bridges the gap between heterogeneous urban data sources and executable co-simulation environments. COSIMO provides a unified machine-readable representation integrating urban data, simulation models, and coupling patterns through a three-graph architecture comprising the Model Catalog, Topologies Catalog, and Data Graph. The framework automates topology validation and instantiation through SPARQL-based reasoning, ensuring physical meaningfulness of connections beyond syntactic compatibility. Through RML mappings and model generation services, COSIMO systematically transforms heterogeneous data sources into simulator-specific formats, reducing the manual effort required for configuration.

Experimental validation across four scenarios confirmed both automation efficiency and physical correctness. Generated co-simulations produced physically meaningful results with energy metrics showing expected behavior patterns corresponding to technology configurations. By enabling domain experts to specify scenarios declaratively rather than configure simulation tools manually, the framework democratizes access to sophisticated multi-energy analysis and supports rapid exploration of decarbonization strategies. This work establishes that ontology-driven approaches effectively automate complex co-simulation construction while maintaining physical correctness and enabling reproducible research, demonstrating that semantic web technologies are ready to support the next generation of urban energy modeling tools required for effective decarbonization planning. Future work will focus on several key directions to enhance COSIMO's capabilities. First, we aim to strengthen topology validation by incorporating comprehensive system design constraints, including numerical stability checks and automated detection of problematic configurations such as algebraic loops, deadlocks, and other patterns that could compromise co-simulation execution. Second, we plan to extend COSIMO toward digital twin capabilities by integrating internet of things data streams directly into the knowledge graph, creating a real-time semantic representation of physical urban systems that continuously synchronizes simulated and observed states. Finally, we intend to leverage artificial intelligence techniques to automate the integration of new simulation models by intelligently matching coupling variables based on semantic descriptions provided by modelers, and by employing dependency resolution strategies to automatically infer appropriate synchronization patterns and coupling configurations, further reducing the manual effort required for co-simulation assembly while ensuring robust and stable system behavior.

REFERENCES

[1] J. Arenas-Guerrero, D. Chaves-Fraga, J. Toledo, M. S. Pérez, and Ó. Corcho, "Morph-KGC: Scalable knowledge graph materialization with mapping partitions," *Semantic Web*, vol. 15, no. 1, pp. 1–20, 2022.

[2] L. Barbierato, P. R. Mazzarino, M. Montarolo, A. Macii, E. Patti, and L. Bottaccioli, "A comparison study of co-simulation frameworks for multi-energy systems: The scalability problem," *Energy Informat.*, vol. 5, no. S4, p. 53, 2022.

[3] R. Battle and D. Kolas, "GeoSPARQL: Enabling a geospatial semantic web," *Semantic Web*, vol. 3, no. 4, pp. 355–370, 2011.

[4] L. Bottaccioli, E. Patti, E. Macii, and A. Acquaviva, "GIS-based software infrastructure to model PV generation in fine-grained spatio-temporal domain," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2832–2841, Sep. 2018.

[5] S. Brandi, P. Rando Mazzarino, D. S. Schiera, M. Bilardo, L. Barbierato, L. Bottaccioli, E. Patti, and A. Capozzoli, "Scenario analysis for renewable energy communities through advanced co-simulation infrastructure," *Appl. Energy*, vol. 394, Sep. 2025, Art. no. 126106.

[6] *Industry Foundation Classes (IFC) Release 4 Add2 Technical Corrigendum 1. buildingSMART International*, Standard ISO 16739-1:2016, 2020.

[7] F. Chini, D. Canali, L. Barbierato, D. S. Schiera, L. Bottaccioli, A. Margara, and E. Patti, "Streamlined deployment of a distributed and scalable co-simulation platform for integrated energy systems," in *Proc. 22nd Int. Conf. Intell. Syst. Appl. to Power Syst. (ISAP)*, Sep. 2024, pp. 1–6.

[8] *Service Vocabulary, 2023*, DINI AG KIM, 2023.

[9] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Comput. Surveys*, vol. 35, no. 2, pp. 114–131, Jun. 2003.

[10] R. García-Castro, M. Lefrançois, M. Poveda-Villalón, and L. Daniele, "The ETSI SAREF ontology for smart applications: A long path of development and evolution," *Energy Smart Appliances, Appl.*, pp. 183–215, 2023.

[11] D. Garijo, V. Ratnakar, Y. Gil, D. Khider, M. Osorio, H. Vargas, and J. Pujara, "Software description ontology for models (SDM)," Tech. Rep., 2020.

[12] C. Gomes, C. Thule, D. Broman, P. G. Larsen, and H. Vangheluwe, "Co-simulation: A survey," *ACM Comput. Surveys*, vol. 51, no. 3, pp. 1–33, 2018.

[13] D. M. L. González and J. G. Rendon, "Opportunities and challenges of mainstreaming distributed energy resources towards the transition to more efficient and resilient energy markets," *Renew. Sustain. Energy Rev.*, vol. 157, 2022, Art. no. 112018.

[14] G. Gröger and L. Plümer, "CityGML–Interoperable semantic 3D city models," *ISPRS J. Photogramm. Remote Sens.*, vol. 71, pp. 12–33, Jul. 2012.
[15] T. Gruber, "What is an ontology?" Dept. Knowledge Systems Laboratory, Stanford University, Tech. Rep., 1993.
[16] T. Hardy, B. Palmintier, P. Top, D. Krishnamurthy, and J. Fuller, "HELICS: A co-simulation framework for scalable multi-domain modeling and analysis," *IEEE Access*, vol. 12, pp. 24325–24347, 2024.
[17] A. Hogan, E. Blomqvist, M. Cochez, C. D'Amato, G. De Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, and S. Neumaier, "Knowledge graphs," *ACM Comput. Surveys (CSUR)*, vol. 54, no. 4, pp. 1–37, Aug. 2021.
[18] W. Huang, P. Olsson, J. Kanters, and L. Harrie, "Reconciling city models with BIM in knowledge graphs: A feasibility study of data integration for solar energy simulation," *ISPRS Ann. Photogramm.*, vol. -4, pp. 93–99, Sep. 2020.
[19] M. Hussain, N. Masoudi, G. Mocko, and C. Paredis, "Approaches for simulation model reuse in systems design—A review," *SAE Int. J. Adv. Current Practices Mobility*, vol. 4, no. 2, pp. 1–19, Jun. 2022.
[20] A. Iglesias-Molina, D. V. Assche, J. Arenas-Guerrero, B. D. Meester, C. Debruyne, S. Jozashoori, P. Maria, F. Michel, D. Chaves-Fraga, and A. Dimou, "The RML ontology: A community-driven modular redesign after a decade of experience in mapping heterogeneous data to RDF," in *Proc. Int. Semantic Web Conf.*, 2023, pp. 152–175.
[21] A. Kelley and D. Garijo, "A framework for creating knowledge graphs of scientific software metadata," *Quant. Sci. Stud.*, vol. 2, no. 4, pp. 1423–1446, Dec. 2021.
[22] T. H. Kolbe, C. Nagel, and J. Herreruela, "3D City Database for CityGML," Tech. Rep., 2013.
[23] M. Lefrançois, A. Zimmermann, and M. Poveda-Villalón, *SAREF Extension for Systems*, Standard ETSI TS 103 548, 2023.
[24] Y. Li, J. Chen, Z. Hu, H. Zhang, J. Lu, and D. Kiritsis, "Co-simulation of complex engineered systems enabled by a cognitive twin architecture," *Int. J. Prod. Res.*, vol. 60, no. 24, pp. 7588–7609, Dec. 2022.
[25] R. Ma, Q. Li, B. Zhang, H. Huang, and C. Yang, "An ontology-driven method for urban building energy modeling," *Sustain. Cities Soc.*, vol. 106, 2024, Art. no. 105394.
[26] M. Mitterhofer, G. F. Schneider, S. Stratbücker, and K. Sedlbauer, "An FMI-enabled methodology for modular building performance simulation based on semantic web technologies," *Building Environ.*, vol. 125, pp. 49–59, 2017.
[27] M. A. Musen, "The protégé project: A look back and a look forward," *AI Matters*, vol. 1, no. 4, pp. 4–12, Jun. 2015.
[28] P. Novák and R. ŠindeláV́, "Applications of ontologies for assembling simulation models of industrial systems," in *Proc. Move Meaningful Internet Syst., OTM Workshops*, 2011, pp. 148–157.
[29] (2024). *OpenWeather API*. [Online]. Available: https://openweathermap.org/api
[30] P. Pauwels and W. Terkaj, "Building element ontology, 2017," Tech. Rep., 2017.
[31] M. Poveda-Villalón and R. García-Castro, "Extending the saref ontology for building devices and topology," in *Proc. LDAC*, Nov. 2019, pp. 16–23.
[32] E. Prud'hommeaux and C. Buil-Aranda, "SPARQL 1.1 federated query," Tech. Rep., 2013.
[33] R. Reda, A. Carbonaro, V. D. Boer, R. Siebes, R. V. D. Weerdt, B. Nouwt, and L. Daniele, "Supporting smart home scenarios using OWL and SWRL rules," *Sensors*, vol. 22, no. 11, p. 4131, 2022.
[34] I. Richardson, M. Thomson, and D. Infield, "A high-resolution domestic building occupancy model for energy demand simulations," *Energy Buildings*, vol. 40, no. 8, pp. 1560–1566, Jan. 2008.
[35] H. Rijgersberg, M. van Assem, and J. Top, "Ontology of units of measure and related concepts," *Semantic Web*, vol. 4, no. 1, pp. 3–13, 2013.
[36] M. Rindarøy, H. Nordahl, S. Sadjina, S. Skjong, and M. Hagaseth, "Adding higher-level semantics to functional mock-up units for easier, faster, and more robust co-simulation connections," *Softw. Syst. Model.*, vol. 24, no. 2, pp. 471–487, 2025.
[37] J. S. Schwarz, R. Elshinawy, R. Acosta, and S. Lehnhoff, "Ontological integration of semantics and domain knowledge in hardware and software co-simulation of the smart grid," in *Proc. Int. Joint Conf. Knowl. Discovery*, 2020, pp. 283–301.
[38] J. S. Schwarz, C. Steinbrink, and S. Lehnhoff, "Towards an assisted simulation planning for co-simulation of cyber-physical energy systems," in *Proc. 7th Workshop*, 2019, pp. 1–6.
[39] S. Aachen, "3D-modell der stadt Aachen," Tech. Rep., 2024.
[40] C. Steinbrink, A. A. van der Meer, M. Cvetkovic, D. Babazadeh, S. Rohjans, P. Palensky, and S. Lehnhoff, "Smart grid co-simulation with MOSAIK and HLA: A comparison study," *Comput. Sci.-Res. Develop.*, vol. 33, nos. 1–2, pp. 135–143, Feb. 2018.
[41] L. Stepien and F. Köster, "Flexible model exchange in modelling smart mobility by using domain ontologies," in *Proc. IEEE Int. Syst. Conf. (SysCon)*, Apr. 2022, pp. 1–7.
[42] M. Stüber, "Models and simulations in the semantic web," Doctoral dissertation, Saarländische Universitäts- und Landesbibliothek, 2023.
[43] M. Stüber and G. Frey, "Dynamic system models and their simulation in the semantic web," *Semantic Web*, vol. 15, no. 2, pp. 353–388, Apr. 2024.
[44] M. Torlini, P. R. Mazzarino, D. S. Schiera, L. Barbierato, L. Bottaccioli, and E. Patti, "An ontology-based methodology to assist complex energy co-simulation setup," in *Proc. IEEE Int. Conf. Environ. Electr. Eng. IEEE Ind. Commercial Power Syst. Eur. (EEEIC/CPS Europe)*, Jul. 2025, pp. 1–6.
[45] *Paris Agreement*, United Nations Framework Convention on Climate Change, 2015.
[46] *RDF 1.1 Concepts and Abstract Syntax, 2014*, W3C, 2014.
[47] *RDF Schema 1.1, 2014*, W3C, 2014.
[48] *OWL 2 Web Ontology Language Document Overview (second Edition), 2012*, W3C OWL Working Group, 2012.
[49] *SPARQL 1.1 Query Language*, W3C SPARQL Working Group, 2013.
[50] Z. Wu, J. C. P. Cheng, Z. Wang, and H. H. L. Kwok, "An ontology-based framework for automatic building energy modeling with thermal zoning," *Energy Buildings*, vol. 296, Oct. 2023, Art. no. 113267.

MATIA TORLINI received the B.Sc. and M.Sc. degrees in computer engineering from the Politecnico di Torino, Turin, Italy, where he is currently pursuing the Ph.D. degree. His main research interests include knowledge graphs, co-simulation, and semantic web technologies for smart energy systems.

PIETRO RANDO MAZZARINO (Member, IEEE) received the B.Sc. degree in energy engineering, the M.Sc. degree in ICT for smart societies, and the Ph.D. degree (summa cum laude) in computer and control engineering from the Politecnico di Torino, Turin, Italy. He is currently a Postdoctoral Researcher at the Energy Center Laboratory, Politecnico di Torino. His main research interests span from urban energy system modeling with a focus on automated data and model extraction to distributed co-simulation of multi-energy systems with a particular focus on intelligent control strategies for enabling energy flexibility and savings.

DANIELE SALVATORE SCHIERA (Member, IEEE) received the B.Sc. degree in energy engineering from the Universitá degli Studi di Palermo, Palermo, Italy, in 2016, and the M.Sc. degree in energy and nuclear engineering and the Ph.D. degree (summa cum laude) in energetics from the Politecnico di Torino, Turin, Italy, in 2018 and 2022, respectively. He is currently an Assistant Professor at the Politecnico di Torino. He has been with Politecnico di Torino, since 2018. His main research interests include the development of holistic methodologies, co-simulation infrastructures, models and toolsets for the deployment of sustainable integrated multi-energy systems.

JULIÁN ARENAS-GUERRERO received the B.Sc. degree in computer science engineering from the Universidad Complutense de Madrid, in 2019, and the M.Sc. and Ph.D. (cum laude) degrees in artificial intelligence from the Universidad Politécnica de Madrid, Spain, in 2020 and 2025, respectively. He is an Assistant Professor with the Universidad Politécnica de Madrid. His research focuses on efficient semantic data management, knowledge graphs, and the semantic web.

LUCA BARBIERATO (Member, IEEE) received the M.Sc. degree in telecommunications engineering from the University of Brescia, Brescia, Italy, in 2013, and the Ph.D. degree (summa cum laude) in computer and control engineering from the Politecnico di Torino, Turin, Italy, in 2022. In 2016, he joined the Department of Computer and Control Engineering, Politecnico di Torino, as a Research Assistant. He is an Assistant Professor with the Department of Computer and Control Engineering, Politecnico di Torino. His primary research interests include energy efficiency in smart cities, with a particular emphasis on designing co-simulation platforms for multi-energy systems.

LORENZO BOTTACCIOLI (Member, IEEE) received the B.Sc. degree in mechanical engineering from the University of Perugia, Perugia, Italy, in 2011, and the M.Sc. degree in energy and nuclear engineering and the Ph.D. degree (cum laude) in computer engineering from the Politecnico di Torino, Turin, Italy, in 2014. He is an Associate Professor with the Energy Center Laboratory, Politecnico di Torino, Turin, Italy. His main research interests include smart energy, cities, and communities, with a focus on software solutions for planning, analyzing, and optimizing smart energy systems, as well as the spatial representation of energy information.

EDOARDO PATTI (Member, IEEE) received the M.Sc. and Ph.D. degrees in computer engineering from the Politecnico di Torino, Turin, Italy, in 2010 and 2014, respectively. He is an Associate Professor with the Politecnico di Torino. From 2014 to 2015, he held an academic visiting position with The University of Manchester. His primary research interests include ubiquitous computing, the Internet of Things, smart systems and cities, distributed software architectures for ambient intelligence, software solutions for simulating and optimizing energy systems, and software solutions for energy data visualization to increase user awareness.