# Real-time Multi-scale Smart Energy Management and Optimisation (REMO) for Buildings and Their District

A thesis submitted to Cardiff University for the degree of

Doctor of Philosophy

by

Bejay Jayan

Supervisors – Dr. Haijiang Li, Prof. Yacine Rezgui

September 2016

## Abstract

Energy management systems in buildings and their district today use automation systems and artificial intelligence (AI) solutions for smart energy management, but they fail to achieve the desired results due to the lack of holistic and optimised decision-making. A reason for this is the silo-oriented approach to the decision-making failing to consider cross-domain data. Ontologies, as a new way of processing domain knowledge, have been increasingly applied to different domains using formal and explicit knowledge representation to conduct smart decision-making. In this PhD research, Real-time Multiscale Smart Energy Management and Optimisation (REMO) ontology was developed, as a cross-domain knowledge-base, which consequently can be used to support holistic real-time energy management in districts considering both demand and supply side optimisation. The ontology here, is also presented as the core of a proposed framework which facilitates the running of AI solutions and automation systems, aiming to minimise energy use, emissions, and costs, while maintaining comfort for users. The state of the art AI solutions for prediction and optimisation were concluded through authors involvement in European Union research projects. The AI techniques were independently validated through action research and achieved about 30 - 40 % reduction in energy demand of the buildings, and 36% reduction in carbon emissions through optimisation of the generation mix in the district.

The research here also concludes a smart way to capture the generic knowledge behind AI models in ontologies through rule axiom features, which also meant this knowledge can be used to replicate these AI models in future sites. Both semantic and syntactic validation were performed on the ontology before demonstrating how the ontology supports the various use cases of the framework for holistic energy management. Further development of the framework is recommended for the future which is needed for it to facilitate real-time energy management and optimisation in buildings and their district.

## Acknowledgements

Though this PhD thesis is a result of many hours of hard work, dedication and perseverance, there are a few people who helped me to lay the foundations to this work and I would like to acknowledge my deepest gratitude to them. In the many ups and downs over the past few years of my PhD journey, these individuals stood by me offering guidance, support and encouragement when necessary for which I am greatly indebted to them.

Firstly, my supervisors Dr. Haijiang Li and Prof. Yacine Rezgui, for providing me the opportunity to carry out this research with Cardiff University and also for their constant guidance and support during these years.

I was truly blessed to have a great team of researchers and colleagues at the BRE institute of sustainable engineering in Cardiff university. Specially would like to thank Dr. Jean-Laurent Hippolyte for his support; most importantly challenging me to bring out the best I could give to this field of research. My sincere thanks also to Dr. Baris Yuce, Dr. Ioan Petri, Dr. Chunfung Yang, Mr. Shaun Howell, and Dr. Muhammad Waseem Ahmad.

Family, I believe was my greatest support who kept me motivated and focused during these years. I would like to specially thank my father for his motivation and inspiration, without which I would not have ventured into the sustainability domain. A big thank you to all my friends from the Jesus Youth Movement for their prayers and support. Above all, I thank my heavenly father, my God. I believe every ground breaking thought and knowledge this research has produced is from him alone. This is his grace.

## Contents

## List of Figures

## List of Tables

## List of Abbreviations

AEC - Architecture Engineering and Construction

AI - Artificial Intelligence

ANN - Artificial Neural Networks

API - Application Programming Interface

BAS - Building Automation Systems

BEMS - Building Energy Management System

BGCBC - Blaenau Gwent County Borough Council

BIM - Building Information Modelling

BMS - Building Management System

BPMN - Business Process Modelling Notation

CCHP - Combined Cooling and Heating Power

CHP - Combined Heat and Power

CIM - Common Information Model

CUSP - Computational Urban Sustainability Platform

DER - Distributed Energy Resources

DHC - District Heating and Cooling

EE - Energy Efficiency

EIR - Employers Information Requirement

EMS - Energy Management System

FM - Facility Management

FP7 - Framework Programme 7

GA - Genetic Algorithm

GHG - Greenhouse Gas Emissions

HPC - High Performance computing

HVAC - Heating Ventilation and Air Conditioning

ICT - Information Communication and Technology

IDM - Information Delivery Manual

IEC - International Electro-technical Commission

IFC - Industry Foundation Classes

ISO - International Standard Organisation

MAS - Multi Agent Systems

MILP - Multi Integer Linear Programming

MINLP - Multi Integer Non-Linear Programming

MVD - Model View Definition

NSGA - Non-Dominated Sorting Genetic Algorithm

nZEB - Near Zero Energy Building

nZEN - Near Zero Energy Neighbourhood

OWL - Ontology Web Language

PMV - Predictive Mean Vote

RDF - Resource Description Framework

RDFS - Resource Description Framework Schema

RES - Renewable Energy Systems

RHI - Renewable Heat Incentive

RSA - Response Surface Approximation Model

SHE - Smart Home Environment

SPARQL - SPARQL Protocol and RDF Query Language

SPARQL Inferencing Notation

SPIN - SPARQL Inferencing Notation

SUMO - Suggested Upper Mereology Ontology

SVM - Support Vector Machine

SWRL - Semantic Web Rule Language

W3C - World Wide Web Consortium

# 1. Introduction

## 1.1. Research Motivation

One of today's greatest challenges is to manage and meet the "*growing demand for secure, affordable energy while addressing climate change and other environmental and social issues*" due to the rising population and increase in economic growth (British Petroleum 2013). Although the global energy demand is expected to increase by 41% between 2012 and 2035, the real challenge lies in how to minimise the level of carbon emissions, as it is expected to increase by 29% during this period. In December 2015, 195 countries worked together to set out the first-ever global action plan to limit the level of carbon emissions and put an agreement in place to limit global warming to well below 2 ℃. (European Commission 2015). This event clearly highlights that the increasing carbon emissions have become a global threat and that every country needs to play its part in containing this issue and help in the reduction of carbon emissions and other greenhouse gas emissions.

Increasing the efficient use of energy can help lower carbon emissions. There is a huge potential to reduce emissions by focusing on the built environment – buildings. Figure 1 below shows that buildings are responsible for almost 35% of the global energy consumption, and therefore any country looking to increase energy efficiency and minimise $CO_2$ emissions needs first to tackle this problem in the building sector (International Energy Agency 2013). In the UK, buildings account for about 37% of total greenhouse gas emissions (Committee on Climate Change 2013). An estimated 77% reduction in $CO_2$ emissions in the building sector compared to today's level would be needed by the year 2050 to meet the goal of limiting the global temperature rise to 2 ℃ (International Energy Agency 2013).

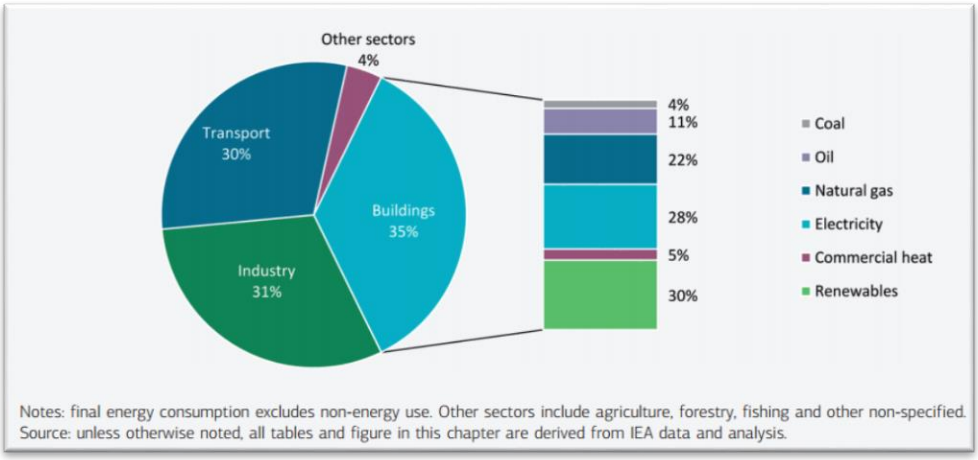



**Figure 1. Final energy consumption by sector and buildings energy mix (International Energy Agency 2013)**

## 'Performance Gap' and 'Multi-objective problems' in Buildings

In addressing the emissions issue in the building sector, one of the key issues that requires attention is the operational management of buildings. It is important to consider operational data because buildings spend most of their time in the operational stages and therefore they need to be managed well to increase operational efficiency. Building management systems (BMSs) are used for this purpose, and they are capable of automatic control of some of the energy consuming operations in a building – such as heating, ventilation or indoor climate. In Europe, appropriate BMS use can save up to 50% of the energy used (Huber et al. 2015). Thus, improving the BMS performance to increase energy efficiency is critical. Failing to do this can lead to what is called a 'performance gap,' where a building is not operating the way it was designed to. One of the reasons for this is the lack of adaptation of the control systems within BMS to the real-time data, which would include adjusting the performance of building energy systems according to the data collected or the user requirements. This process can be costly and hence often is neglected by owners/clients. Therefore, an automated and intelligent self-learning system built to increase the energy efficiency of buildings based on operational data would be ideal in today's reality.

Energy management in buildings should minimise the energy demand and emissions while keeping the costs low. This process is becoming more complex than before, especially for contemporary buildings, due to the multi-domain and multi-objective nature of the problem. Therefore, a large amount of relevant information (from different data silos) needs to be considered and processed (Schellong 2012) for optimum control and decision-making. Holistic decision-making requires the consideration of more interlinked data and information (Irani et al. 2015). For example, providing weather information will help forecasting tools to predict the demand accurately. This demand can be further reduced by optimising the performance of energy using systems. Holistic and intelligent energy management systems are needed today that ensure the highest operational efficiency while satisfying the various objectives. 'Holistic' here means considering not just the objectives within the building domain but also their interactions in a larger context, as explained below.

**The need for energy management at a district level**

Looking at buildings alone cannot achieve holistic and efficient energy management. One of the reasons for this is that, today, decentralised systems (Distributed Energy Resources (DERs)) are increasingly adopted as an alternative to centralised energy sources or as an additional energy supply to the main grid. These systems provide distributed energy at a smaller scale, such as cities and districts (Bazmi and Zahedi 2011). Today, cities and districts stand at the forefront of the sustainability agenda because they are major consumers of energy and resources (Wang and Prominski 2016). Thus, buildings and their interactions in a wider context should be considered for holistic energy management (Allegrini et al. 2015). A new integrated approach of interconnectivity between buildings, DERs, grids and other networks at a district level is required. This thesis defines a district using the following two criteria: (1) Has two or more buildings as energy consumers in the vicinity; (2) Relies on decentralised energy systems or local sources of energy (heat and electricity). In certain cases, the district can depend on a central source as well to meet a part of its energy demand.

Current decentralised systems include co-generation technologies such as combined heat and power (CHP), using biomass power, solar PV power, wind power, and so forth at a local or regional level. District heating and cooling (DHC) systems are increasingly being used in districts today (Euroheat and Power 2014); they produce steam and hot/chilled water in a central plant and distribute this to individual buildings (residential and commercial) in their vicinity through a network of pipelines. DHC systems have been widely utilised in hospitals, industrial parks, office complexes, large campuses (universities), housing estates and small districts, which can also have a mix of the above buildings due to the benefits of saving energy, consumer space and inhibiting air-pollution (Sakawa et al. 2001). DHC systems using CHP are increasingly becoming a popular choice not just in Europe, but in many other countries such as the United States, China, Russia, and India. They are very effective in reducing greenhouse gas emissions

(GHG) and increasing economic benefits (Jamot and Olsson 2013). DHC networks are also a long-term asset, according to the International Energy Agency, as they are a bridge towards the future low carbon energy technologies (International Energy Agency 2009). For example, they are capable of taking heat from any source including renewable heat sources, hence offering flexibility to integrate new low-carbon sources when made available in the future (Jamot and Olsson 2013). However, the development of district energy systems – in particular, Renewable Energy Sources (RES) – requires new business and technology platforms to manage the increased level of complexity and diversity of global energy management.

Managing and improving operational energy efficiency in buildings and their district can be challenging due to the following factors:

(1) The integration of renewables and the use of co-generation plants in today's energy mix making the problem more dynamic, uncertain, and complex; and

(2) Many different constraints need to be factored in at each stage of the optimisation, and this requires high computational power to provide near real-time results.

The above need to be addressed to achieve maximum energy efficiency during the complex and dynamic operation stage. Moreover, existing district energy systems can be further improved to achieve better efficiency while using primary energy resources further to reduce their environmental impact (Gadd and Werner 2014). However, achieving this can be a complex task because there can be many energy resources within a district and the decision on which source to use at what time depends on factors such as climate, energy market, user preferences, system efficiencies, maintenance and so forth. Hence Lund et al. (2014) believe that smart energy management techniques which can take these factors into account are needed today. Information and communication technologies can provide solutions which address some of these complex challenges related to energy management (Irani et al. 2015). In summary, what is needed today is a holistic energy management approach considering both demand side (buildings) and supply side (districts) operations which is capable of solving complex multi-objective problems in these domains.

**Energy management today and the need for semantic models**

Figure 2 below shows the hierarchy of solutions adopted today for energy management. The hierarchy here represents the order in which energy management solutions are implemented today. The most basic solutions available today is the implementation of BMS/EMS systems.

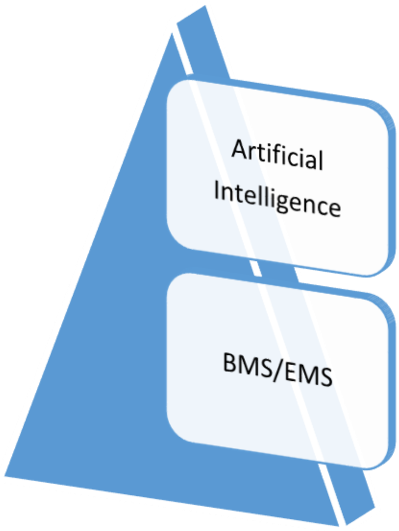



**Figure 2. Techniques applied for energy management today**

BMS in a building or EMS in the district are widely used today to control the day-to-day actions of actuators or devices in buildings and districts. Facility managers usually operate them, but at times operations can be automated by implementing best practice knowledge in these systems. These systems help in detailed monitoring of the performance of buildings and districts as well by recording data from various sensors, meters, and devices.

Artificial intelligence solutions come next in the hierarchy shown in Figure 2, and they are increasingly being used today to add intelligence to the energy management in buildings and districts. Artificial intelligence usually works with automation systems for real-time implementation wherein it uses optimisation algorithms, prediction algorithms, etc., to aid human decision-making. Different types of optimisation techniques have been used, which include single objective or multi-objective algorithms. Multi-objective optimisation is useful to make decisions when there are two or more conflicting objectives. These optimisation algorithms can not only be used to optimise control of energy systems, but also help in high-level planning – such as optimisation of schedules (both demand side and supply side schedules). More recently, two or more of these AI techniques have been combined and used for applications. For example, ANN models forecast the energy demand, consequently allowing optimisation applications to use these results and plan ahead, making it a proactive approach rather than a reactive one.

Both these approaches can lead to an increase in energy efficiency. However, they have their shortcomings when it comes to applications for holistic energy management. For example, automation systems do implement decisions in real time (during operational stages), but they can only deal with data silos and are not supported by cross-domain data to consider multi-objectives of the problem. Artificial intelligence, on the other hand, deals with the issue of multi-objective problems. However, it fails to increase energy efficiency at both demand side (within buildings) and supply side (in district) simultaneously. For example, AI applied to optimise the building demand does not consider the knock-on effect of this on the supply side for the district. Vice versa, applications looking to manage efficiently the energy production in districts (supply side) give less importance to building demand optimisation.

This PhD research therefore investigates the application of semantic models to bridge the gaps highlighted above. Figure 3 below represents the basic idea behind semantic models.

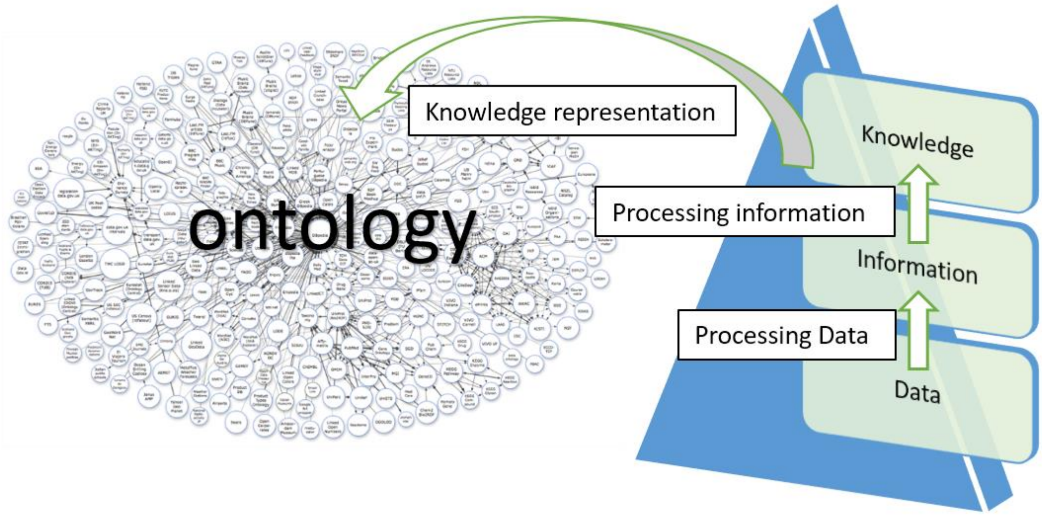



**Figure 3. Ontology representing knowledge**

The figure shows that processing data can give information, and adding meaning to this information gives knowledge. Semantic models are used to capture this knowledge and make them machine-readable. Semantic meaning is added to data models for them to become semantic models. Ontologies as shown above in figure 3 represent semantic models and are usually used to define common vocabulary when researchers need to share information in a domain. Moreover, the basic concepts in the domain and their relationships can be defined here, which is machine interpretable (Noy and McGuinness 2001).

One of the major reasons for using semantic models in this research is that they *"can be branched across domains of knowledge automatically"* (Linked Data Tools 2015). In other words, a semantic model can bring heterogeneous domains together and can be consequently used as a knowledge base to support particular applications. Using this advantage of ontologies, the author through this research aims to build a knowledge-base which links knowledge from the demand and supply sides together. Consequently, the author looks to use this knowledge to support the functioning of AI and automation systems.

A large variety of disciplines develop standardised ontologies through which information can be shared and annotated by domain experts. The medical field widely uses ontologies, where a lot of large, standardised, structured vocabularies have been developed (Noy and McGuinness 2001). Broad general-purpose ontologies are also being developed here, for example, the UNSPSC (United Nations Standard Products and Services Code) ontology[1] developed provides terminology for products and services.

In the field of energy management, however, domain-independent applications, software agents, or problem-solving methods use ontologies; and at times they are not used merely as a means to share a domain model. For example, in buildings, more of these ontologies are built as a knowledge-base, which consequently facilitates intelligent energy management for rule-based decision-making. Recent years have seen more European research projects (under Framework Programme 7 (FP7) research) adopting solutions that implement ontology-based systems for decision-making.

## 1.2. Problem Statement

In the case of districts today, demand side management in buildings and supply side management at the district level do not work together holistically. Applications looking to manage energy production in districts (at the supply side) normally do not include details within the buildings (demand side) and the energy management decisions taken here. Moreover, energy management in each of this domain are complex in nature which needs to consider various data domains and factors. To deal with this issue of holistic intelligent energy management, this research develops an ontology taking into account the multi-scale (buildings and districts) and multi-objective nature of the problem. Ontology developed in this research represent inter-linked knowledge across buildings

[1] http://www.ebusiness-unibw.org/ontologies/pcs2owl/

and districts, aiming to support the holistic decision-making framework with information needed.

### 1.3. Research Aim and Objectives

The aim of this research is to develop and validate an ontology which supports the seamless integration of artificial intelligence (AI) techniques and automation systems, dynamically mapping demand side and supply side energy, enabling 'holistic' decision-making taking into account the various objectives involved. The study also investigates how best to capture the knowledge behind the artificial intelligent models (such as optimisation and prediction models) in the ontology for replication in similar sites for future projects.

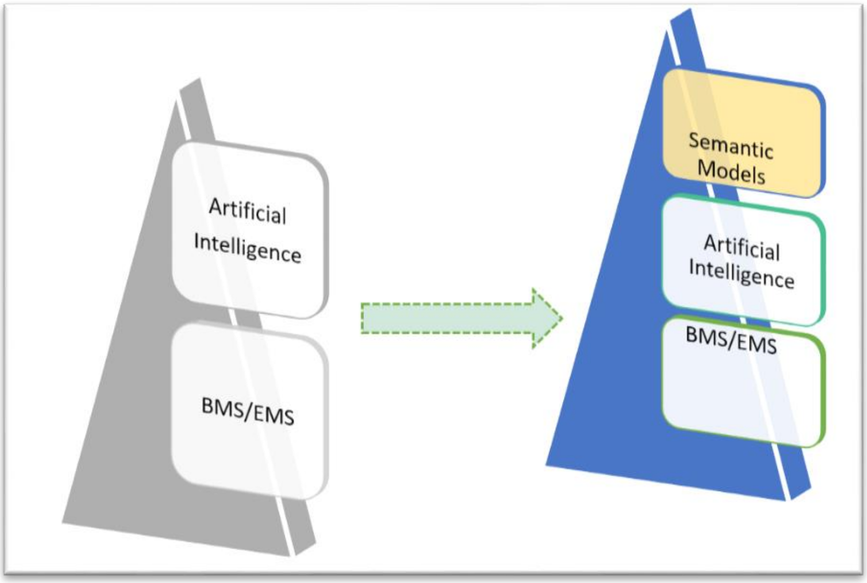



**Figure 4. Solution for holistic energy management**

Figure 4 above, therefore, depicts the transition needed today from the author's point of view when compared to Figure 2 presented earlier. A triple-layered system is needed today to achieve holistic real-time energy management. The system represents the shape of a pyramid, where the higher the level of implemented solution, the higher the energy efficiency. It highlights that automation systems are the foundational requirement for any real-time energy management solutions to be applied in buildings or districts. Semantic models, on the other hand, in figure 4 is needed to add intelligence to existing methods, making it more efficient as it takes a holistic approach.

**Research Objectives**

1. To determine the state-of-the-art artificial intelligence techniques for real-time energy management for demand side management and supply side management.
2. To harmonise the real-time energy management solutions developed at demand side and supply side.
3. To develop and validate an ontology which support the use of AI models with automation systems. It should also support the harmonisation of demand and supply side energy management.
4. Determine how to capture the knowledge behind the generation of these AI models in the ontology developed and reuse it for similar buildings and districts in the future.

## 1.4. Hypothesis and Research Questions

Ontologies can be used to support the seamless integration of numerical optimisation algorithms and automation systems through a cross-domain knowledge-base, consequently, this can be used to aid smart decision making processes autonomously for building and district energy management. The knowledge captured in the ontology can also further facilitate the autonomous definition of Artificial Intelligence models needed for smart decision-making.

**Research questions**

- What are the artificial intelligence techniques applied at a building and district level for real-time energy management?
- What are the various objectives that need to be considered for energy management when it comes to building and district energy management?
- How can building and district energy management techniques be harmonised to take a multi-objective approach?
- Can ontologies help facilitate the harmonisation of demand and supply side optimisation? Moreover, how?
- Can the knowledge behind the optimisation models be captured in the ontology so that they can be replicated for similar districts and buildings?

## 1.5. Research Contribution

This thesis presents an ontology which captures the knowledge needed to support the running of a smart decision-making framework that can be applied to districts and their buildings, solving multi-objective optimisation problems for energy management in real time. The fact that this proposed decision making framework can be applied in the operational stages, taking into account the dynamic changes in the supply and demand sides, is very innovative. The research also proposes a holistic approach to decreasing the energy demand and dynamically mapping the energy production (supply side in districts) to this reduced demand, to increase the overall energy efficiency.

Although the research focusses on the ontology development and its validation, it illustrates how the proposed ontology-based framework can capture and orchestrate the running of artificial intelligence models, e.g. numerical optimisation, and artificial neural networks, together with automation systems. The ontology developed successfully captures the knowledge behind the use of optimisation models and energy simulation/prediction models, to streamline the reuse of existing knowledge.

## 1.6. Structure of the thesis

Chapter 2 below reviews the literature looking into both artificial intelligence techniques applied to buildings and district energy management. It highlights the possible gaps in the domain and, following this, Chapter 3 then discusses the research methodology adopted to address these gaps. Chapter 4 describes the action research and concludes how this knowledge has helped for the system design and implementation. This consequently leads on to Chapter 5, describing the overarching system framework and its various modules. The ontology design and the various functionalities are discussed in detail, as this is the core of the framework. Chapter 6 presents the system implementation and development aspects of the framework. Chapter 7 further presents the validation of the ontology and finally, Chapter 8 summarises the major conclusions and contributions of the research along with the future work.

# 2. Literature Review

This chapter assesses previous work in the field of real-time energy management in buildings and districts. The review covers different techniques, methods, and algorithms used for energy management. The chapter is mainly divided into two parts – Section 2.1 reviews optimisation algorithms and AI solutions applied previously in this domain whereas, Section 2.2 looks into ontologies applied previously.

## 2.1. Optimisation algorithms and AI applied today

This section reviews some of the AI solutions applied in the field of energy management and optimisation, for both the building and district domain. Background theory on some of the most relevant AI techniques are presented below in section 2.1.1, before reviewing some its applications in the building and district domain in Section 2.1.2. and Section 2.1.3. respectively. Section 2.1.4 discusses some of the key findings from this part of the review.

### 2.1.1. Background Theory

*Optimisation theory*

Optimisation is an iterative process to search for a solution that minimises values of the objective function while satisfying the constraints imposed on design variables and the system responses. A generic formulation for an optimisation problem is given as follows:

$$min \rightarrow f_0(x)$$
$$subject\ to\ f_i(x) \leq 0, i = 1, \cdots, m$$
$$h_j(x) = 0, j = 1, \cdots, p$$

- vector x $\in R^n$ represents the design variables;
- the function $f_0: R^n \rightarrow R$ represents the objective function. The objective function can be either mathematical functions of the design variables or even black box problems.
- the functions $f_i: R^n \rightarrow R, i = 1, \cdots, m$ represent the inequality constraint functions,
- the functions $h_j: R^n \rightarrow R$, j $= 1, \cdots, p$ represent the equality constraint functions.

A vector $x^*$ can be the optimal solution of the problem, provided it has the least objective value among all vectors that satisfy the constraints. The optimisation process is generally iterative, which begins with initial values of the design variables, and consequently generates a sequence of estimates (estimation design points) for the design variables. The optimisation process ends when a design point reaches a solution or meets the terminate criteria.

*Multi-objective optimisation*

A minimisation multi-objective optimisation problem with Q objectives is listed below (Konak et al. 2006):

$$x = \{x_1, x_2 \dots x_n\}$$

Here, $x$ is an $n$-dimensional decision variable vector in the solution space X. A set of Q objective functions that needs to be minimised can be defined as:

$$\mathrm{z}(x^*) = \{z_1(x^*), z_2(x^*) \dots z_Q(x^*)\}$$

Here, $x^*$ vector is the solution to minimisation of the objective functions. All objectives are defined as the minimisation type, and it can be converted to maximisation type by multiplying by -1. A series of constraints is possible in this problem:

$$g_i(x^*) = b_i \text{ for i} = 1, \dots, \text{m}.$$

There can also be bounds on the decision variables. In most energy-related problems, it might be necessary to satisfy more than one objective. These objectives can be conflicting, which means finding an optimised solution for $x$ minimising one objective can compromise the other objective(s). Therefore, a reasonable solution to the multi-objective problem would be to find a set of solutions, each of which can satisfy every objective to an acceptable level and is non-dominated by any other solution.

A feasible solution $x$ dominates another feasible solution $y$ ($x$>$y$), if $z_j(x) \leq z_j(y)$ for $j = 1, \dots,$ Q, and $z_i(x) < z_i(y)$ for at least one objective function $i$.

A solution that is non-dominated by any other solution in the solution space is called a Pareto optimal set. However, each solution is different in the Pareto, which means a gain in one objective(s) comes with a sacrifice in the other(s). In this case, the final decision lies with the decision-maker, who decides on the trade-off needed, and hence this is a very practical way to solve real-world problems.

*Genetic algorithms*

Some of the optimisation algorithms are based on a stochastic search approach, such as evolutionary algorithms, simulating annealing, and genetic algorithm. Traditionally, genetic algorithms are known to be better at solving problems. They use specialised fitness functions and promote solution diversity, which makes them capable of accommodating multi-objective problems. Here, a fitness function is a particular objective function that characterises the problem and measures the closeness of a given solution to the target, also considering all the problem constraints (Openeering, n.d.). A GA is a meta-heuristic algorithm which is inspired by the theory of the origin of species. In nature, unfit species within an environment face extinction by natural selection, whereas stronger ones survive and, through reproduction, pass on their genes to future generations. In the longer run, these strong genes become dominant in their population. The genes also evolve through time and, if these changes support them in the challenge of survival, they tend to give way to new species, and unsuccessful changes are eliminated by natural selection.

In GA terminology:

- Each solution vector *x* is called an individual or chromosome.
- Genes are discrete units that make up chromosomes.
- Population is the collection of chromosomes that usually works with GAs.

During the optimisation process, the solution gets fitter as the search evolves. It uses two operators to bring a natural evolution to the chromosomes and generates new solutions from previous ones:

- *Crossover:* here, two chromosomes (parents) combine to form a new chromosome (offspring). Parents are selected from the existing chromosomes that have a preference towards the fitness. This means that the offspring inherits good genes. With more and more iterations applying crossover, more chromosomes with better genes are produced, which eventually converges to a good overall solution.
- *Mutation:* this is implemented at the gene level, where a random variation is brought to the characteristics of the chromosomes. Mutation rates in a GA are very small, and this brings genetic diversity to the population and helps the search look beyond the local optima.

A selection procedure is applied during reproduction of the new population. Usually, the probability of selection is dependent on the fitness of the chromosomes. Some of the popular selection methods are proportional selection, ranking, and tournament selection. Figure 5 below shows the pseudo code for a GA.

```
Input: Population_size, Problem_size, P_crossover, P_mutation
Output: S_best
Population ⟵ InitializePopulation(Population_size, Problem_size)
EvaluatePopulation(Population)
S_best ⟵ GetBestSolution(Population)
While (¬StopCondition())
   Parents ⟵ SelectParents(Population, Population_size)
   Children ⟵ ∅
   For (Parent_1, Parent_2 ∈ Parents)
      Child_1, Child_2 ⟵ Crossover(Parent_1, Parent_2, P_crossover)
      Children ⟵ Mutate(Child_1, P_mutation)
      Children ⟵ Mutate(Child_2, P_mutation)
   End
   EvaluatePopulation(Children)
   S_best ⟵ GetBestSolution(Children)
   Population ⟵ Replace(Population, Children)
End
Return (S_best)
```

**Figure 5. Pseudo code for a genetic algorithm (Brownlee 2015)**

GAs are well suited to solving a multi-objective problem as it is a population-based approach. The GA approach can identify multiple sets of non-dominated solutions in a single run. It is also capable of searching different areas of the solution space simultaneously and providing a diverse set of solutions for complex problems with non-convex, discontinuous and multi-modal solution spaces. Crossover operators exploit structures of good solutions with respect to different objectives and hence give solutions from unexplored areas of the Pareto front. In GAs, it is also not necessary to give weights to the objectives (or prioritise any over the others), and hence it has become the most popular heuristic approach for multi-objective problems.

Out of the many genetic algorithms used for multi-objective problems, NSGA (Non-Dominated Sorting Algorithm) is a popular choice and was developed by Prof. Kalyanmoy Deb (Deb et al. 2002). It classifies the population into non-dominated fronts. NSGA-II is a better algorithm than NSGA as it is more efficient and faster in sorting the non-dominated fronts. These two algorithms are designed to solve non-convex and non-smooth single and multi-objective optimisation problems.

Some of the features of NSGA-II are (Openeering n.d.):

- All individuals are sorted according to the level of non-domination into a hierarchy of non-dominated Pareto fronts.

- Elitism is implemented, where the selection procedure only includes individuals that are non-dominated solutions, and hence enhances converging properties.
- NSGA-II also uses a crowding-distance approach through which fitness-sharing parameters aim to achieve a uniform spread of solutions along the best-known Pareto front.

Figure 6 below shows the pseudo code for the NSGA-II algorithm.

```
Initialize Population
Generate N random solutions and insert into Population
for (i = 1 to MaxGenerations) do
    Generate ChildPopulation of size N
    Select Parents from Population
    Create Children from Parents
    Mutate Children
    Combine Population and ChildPopulations into CurrentPopulation with size
     2N
    for each individual in CurrentPopulation do
        Assign rank based on Pareto – Fast non-dominated sort
    end for
    Generate sets of non-dominated vectors along PF known
    Loop (inside) by adding solutions to next generation of Population starting
     from the best front
        until N solutions found and determine crowding distance between
          points on each front
end for
Present results
```

**Figure 6. Pseudo code for NSGA-II (Syberfeldt 2014.)**

*Artificial Neural networks theory*

Networks are one way in which a complex problem can be broken down into simpler problems which make it easier to comprehend. Networks follow the lemma divide and conquer principle (Bar Yam, 1997). There can be different types of networks, but they are characterised by a set of nodes, and the connections between them. Nodes are computational units, which receive inputs and consequently process them to give outputs, whereas the connection between them represents the information flow, which can be unidirectional (information flows only in one direction) or bidirectional (information flows in either direction). The interaction between these nodes leads to a global behaviour of the network which can be termed as emergent. This behaviour cannot be observed in the individual elements of the network and hence the abilities of the network supersede the abilities of the elements (Carlos Gershenson 2003).

Artificial neural networks see the nodes in the network as 'artificial neurons'. These are computational models inspired by natural neurons which receive signals through synapses (located on dendrites or membranes of the neuron), and, when these signals are strong enough (or surpass a threshold), the neuron is activated, and it emits a signal through the axon. Signals can then be sent to another synapse which again can activate other neurons in the network (Carlos Gershenson 2003). See Figure 7 below.

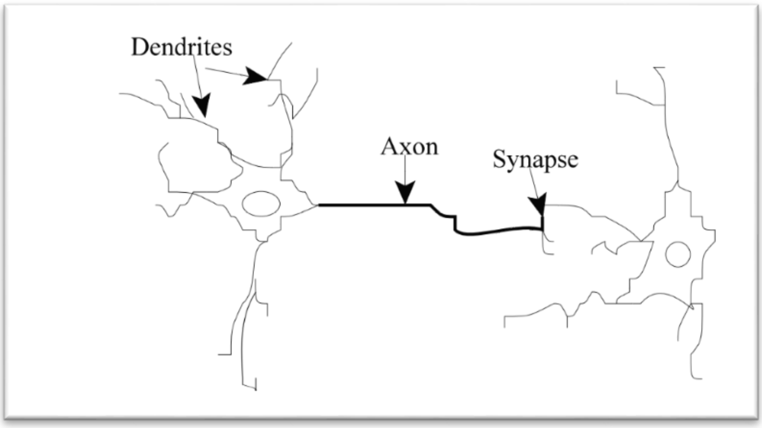



**Figure 7. Structure of a natural neuron (Carlos Gershenson 2003)**

ANNs contain input (like synapses) which is multiplied by weights (to represent the strength of the respective signals). A mathematical function then computes this and determines the activation of the neuron. Another function then computes the output of this. ANNs use these neurons to process information.

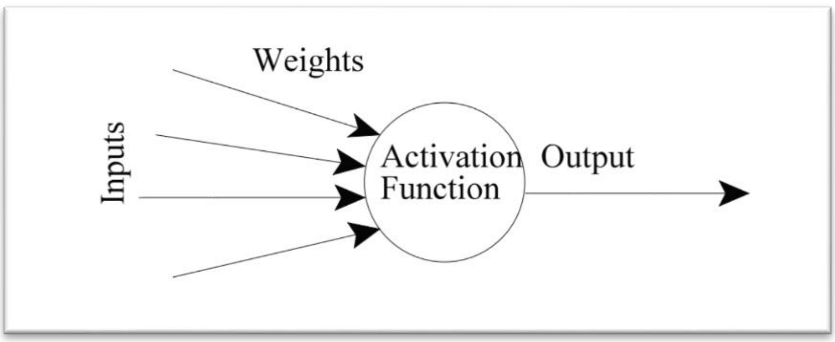



**Figure 8. Activation function and weights for an ANN (Carlos Gershenson 2003)**

The weights determine the computation of the neuron. Adjusting weights of the artificial neuron gives the desired output for a specific input.

However, an ANN network sometimes can have hundreds of neurons and adjusting weights for each can be complicated. Algorithms, therefore, are available that can automatically adjust weights to derive desired outputs, a process which is called learning or training. Learning and training can be very useful when the relationship between input and output variables is complex and hard to formulate. ANN models can act as black boxes, provided sufficient input and output data is available for training them. For this precise reason, ANNs are applied in complex energy-related problems of the building domain to represent complex relationships here as a black box. Once the model is trained, it can then be used as an analytical tool, replacing mathematical models or simulation models, for example.

There can be various possible training algorithms, one of the most common ones being backpropagation algorithm. Backpropagation is an abbreviation of 'backward propagation of error'. This algorithm runs using the delta rule, wherein learning is a supervised process. Here, when the input is presented, it makes a random guess as to what the output might be. Following this, calculation is performed on how far the actual answer is from the known outputs, according to which the connection weights are adjusted (University of Wisconsin, n.d.).

ANN models are arranged in layers of interconnected neurons. The patterns are presented to the network as the input layer, which then communicates to one or more hidden layers where processing is carried out through the weighted connections. The hidden layer then links to the output layer, which represents the output of the network.

### 2.1.2. Building Domain

*Multi-objective optimisation*

Energy management in buildings requires dealing with multi-objective problems with conflicting targets – minimising energy consumption, minimising $CO_2$ emission, maximising comfort level. In practice, this can be challenging and complex to achieve as it depends on various factors within the building environment. Diakaki et al. (2008) and Diakaki et al. (2010) investigated the feasibility of developing a stand-alone multi-objective optimisation model that would help in decision-making when it comes to choosing from many alternative solutions to meet the objectives. The problems investigated here were analysed to keep both cost and energy at a minimum, which is always a dilemma. Also, this methodology does not complement any other method such as simulation, multi-criteria decision analysis techniques and the like. In their later work, Diakaki et al. (2010) did use envelope-related variables and constraints, and also systems-related (heating system, cooling systems, solar collector systems) variables and constraints in their decision model. The objectives set here were clearly defined and formulated into linear or non-linear mathematical expressions.

Many previous works investigated multi-objective optimisation but at the design stages. Hamdy et al. (2011) used MATLAB linked with a multi-objective genetic algorithm (GA) to investigate energy-saving measures by examining 24 design variables which were related to optimal HVAC (Heating Ventilation and Air Conditioning) settings, night ventilation, night set-back temperature, daylighting, etc. Wang et al. (2005) developed an object-oriented framework for simulation-based green building design optimisation using GA again. Bichiou and Krarti (2000) considered three different optimisation algorithms including GA to optimise HVAC system selection for residential buildings. Various parameters and possible values for each are considered here during the optimisation. Wright et al. (2002) demonstrated how a multi-objective GA can be used to find an optimum solution when thermal comfort and cost are taken as objectives. Ihm and Krarti (2012) tried to improve energy efficiency in buildings in Tunisia while keeping the cost low, by examining various design feature combinations including glazing type, location, window sizes, appliances, lighting fixtures, heating and cooling systems, etc. The work mentioned above shows the potential of GA algorithms to solve multi-objective problems, but doing this in real time during the operational stages is important, as referred to in Chapter 1, to reduce the performance gap.

Operational knowledge and data monitoring are important to help reduce the performance gap. The operational data can be used with optimisation models, consequently providing optimum values to control the facility. One way of doing this, therefore, is to integrate these artificial intelligence techniques with existing automation systems or building energy management systems (BEMS). However, BEMS applied nowadays tend to be isolated and lack flexibility, scalability, and adaptation capacity. They fail to take a holistic approach when it comes to the complex multi-objectives, and they more or less take a passive approach to decision-making, failing to respond to the changing environment. For example, most of the control today is based on best practice rules and operational principles which are pre-defined from previous experiences of the facility manager. Therefore, BEMS have to work smartly with AI techniques to produce better energy management systems.

AI techniques used for optimisation needs to be automated when integrating with BEMS. Also, traditional rules of thumb or trial-and-error processes are not sufficient or applicable for building-related multi-objective problems (Bazjanac 2008). Optimisation algorithms are widely classified as conventional gradient-based methods and gradient-free methods. Gradient-based methods involve using mathematical procedures, which are applied to smooth and continuous objective functions. For most building studies, gradient-based optimisation methods are not applicable as their behaviours are often nonlinear and discontinuous (Wetter and Wright 2004). Hence, gradient-free methods, in this case, are more suited. The Genetic algorithms (GA) and their various adaptations are a widely accepted method under this category (Holland 1992).

*Artificial neural network methods*

The optimisation above is implemented on mathematical models which in the case of buildings are simulation models of the facility. Simulation models are needed for calculation or study of energy demand patterns. These models are then used as cost function by the optimisation model to optimise certain decision variables (simulation model input) while meeting the various objectives of the problem. Some of these objectives can be minimising energy, minimising costs, minimising emissions, and maximising comfort.

Building energy simulation-based optimisation, however, can be time-consuming due to the complexity of the models. Most of the existing simulation programs such as EnergyPlus or Trnsys rely on heavy computation time, and repeatedly running them for optimisation would lead to a greater overall computational time to provide optimum results. High computational time is not feasible when the solutions are needed for real-time energy management. In this case, simplifying simulation models to decrease the computational time might mean risking a loss of accuracy. Simulation models, however, can be replaced by surrogate models for prediction or calculation of the energy consumption. Many surrogate models have been investigated previously, as described below.

Zhao and Magoulès (2012) conducted a review of various methods to predict building energy consumption. Accurately predicting the performance of buildings can be a complex task due to different reasons, as hinted by the authors – ambient weather conditions, the complexity of building characteristics (structure and envelope), dynamic changes in occupancy, HVAC system operations, and secondary-level components such as lighting, shading, electrical equipment, etc. The study relates to reviewing the different developed models (statistical methods, simplified engineering methods, and artificial intelligence methods) that help improve the accuracy of prediction. The authors conclude that artificial intelligence methods such as support vector machines (SVMs) and artificial neural networks (ANNs) are the best way forward. SVMs are said to be a better option than artificial neural networks mainly because they can do a similar job to ANNs with less training data. However, they have a lower running speed than ANNs.

ANNs have been widely implemented. They are used in forecasting and prediction of loads (load forecasting, energy management). Kalogirou and Bojic (2000) conducted a general review on the various applications of ANNs, especially focusing on their role in energy systems – predicting a building's thermal load, predicting airflow in a naturally ventilated room, prediction, evaluation, and optimisation of a building's energy consumption. The ANN method has also been used to model the power consumption of a central chiller plant, including chillers, cooling towers and pumps (Yalcintas and Akkurt 2005). The input variables here mainly consider climatic data (dry bulb temperature, wet bulb temperature, dew point temperature, relative humidity percentage, and wind speed and wind direction), and chiller-plant power consumption was chosen to be the output. Ekici and Aksoy (2009) predicted heating loads in buildings by training a neural network with three different building samples, achieving average accuracies of 94.8% - 98.5%. Neto and Fiorelli (2008) compared ANN models and EnergyPlus simulations (with real measured data) to compare their forecasting capabilities for building energy consumption; and concluded that, even though a feed-forward ANN model might not be the most suitable one for building applications, it still has an advantage over EnergyPlus simulations. This study again, like mentioned previously, focused more on climatic data variables like minimum & maximum outdoor air temperature, global solar radiation, and relative humidity. The day type was also considered (i.e. weekend or weekday).

ANNs have been used to evaluate different building simulation packages themselves. Yezioro et al. (2008) used an ANN to predict a whole year's energy consumption data based on acquired data from just a week. The predicted data showed good fitness with the mathematical model, with a mean absolute error of 0.9%. This data was then used to evaluate building simulation packages like eQuest, Green Building Studio, EnergyPlus, and Energy_10. The ANN model input independent variables in this study once again focused largely on climatic data – outdoor temperature, relative humidity, and also on setpoint temperature and occupancy schedule.

ANN models were used to optimise the performance of HVAC systems and in smart decision-making. Ben-Nakhi and Mahmoud (2004) applied general regression neural networks to optimise HVAV thermal storage by predicting cooling load profile based on historical data. This study proved the effectiveness of neural networks even with a single input variable (external temperature). The model was trained using three years of past data collected from three different types of office buildings. Thus, the results produced during prediction were more than satisfactory. Hu and Olbina (2011) developed an illuminance-based slat-angle selection model based on an artificial neural network and used this to predict illuminance at two sensor points in a room. This was then used with a mathematical model to find the optimum slat angle. The results showed that the ISAS model could predict illuminance with an accuracy of 94.7% and optimum slat angle with an accuracy of 98.5%. Ayata et al. (2007) produced ANN models to predict indoor air velocity distributions for natural ventilation.

ANN models have more recently been used in combination with other AI techniques such as optimisation. Magnier and Haghighat (2010) used a simulation tool – Trnsys – and genetic algorithms to find the optimised solution for a problem aiming to satisfy conflicting objectives – thermal comfort and energy consumption within buildings. The study also implements an ANN as the response surface approximation model (RSA), mainly to reduce the computational time that the GA usually takes. The Trnsys-based simulations were validated using measured data, and consequently a database of cases was generated, which was used to train and validate the ANN. This ANN model was then used to identify potential solutions with changing input variables. Results showed that ANN could predict output with good accuracy and also significantly reduce the computational time involved while using a GA. The variables considered in this study were related to HVAC system settings (heating and cooling setpoints, relative humidity setpoints, and supply airflow rate), thermostat programming (starting and stopping delay) and passive solar design (window size and thickness of concrete).

One of the most important studies was conducted by Ben-Nakhi and Mahmoud (2011). They demonstrated that ANNs can be used to predict energy consumption for shorter time periods, i.e. hourly or daily with high accuracy, provided the right variables are chosen. This means that real-time energy optimisation is possible if these prediction models can work with optimisation models. Yang et al. (2005) looked into different training techniques for adaptive neural network model, and consequently used the model for real-time on-line building energy prediction. Adaptive neural network models are special because they can adapt themselves when there is a change in input patterns. However, the major challenge with an ANN is that: 1. It depends heavily on historical data, and 2. It cannot measure energy savings in retrofit strategies because of lack of measured data.

*Summary*

In summary, ANNs have a wide variety of applications in buildings, one of the most common ones being predicting of load or demand. Here, most of the authors relied on indoor and outdoor weather data for training their models. Although not often, ANNs have been used for short-term forecasting applications. They can be used to improve energy efficiency in buildings when combined with optimisation models, as shown briefly in some of the previous works. GA might be the way forward for optimisation, but it is incomplete without these neural networks, which significantly improve computational time and do not compromise the accuracy. The majority of the optimisation work reviewed here shows the huge application in design stages, and the rare application in the operational stages of buildings.

The pre-requisite, however, to using ANN models is to have adequate data for training of the ANN, which would ensure that the black box model is stable and achieves satisfactory results without compromising on the accuracy of the results/model. Using ANN models as cost function of the optimisation problem is well-suited for real-time applications as the evaluation times become negligible. Simulation models can also be used to provide data for calibration or training for new ANN models if historical data is not available.

More recently, EU projects have looked into using ICT (Information and Communication Technology) for increasing energy efficiency in buildings under the FP7-2011-NMP-ENV-ENERGY-ICT-EeB call, including BEAMS (285194), SEAM4US (285408), CASCADE (284920), Campus21 (285729), SEEDS (285150), and KnoholEM (285229). Most of the solutions adopted in these projects investigate using building information modelling techniques with real-time sensing capabilities for the optimisation of energy consumption. This again proves that data (historical or real-time) is critical to today’s energy management solutions and AI solutions.

### 2.1.3. District domain

As mentioned in Chapter 1, it is important to consider both building (demand side) and district (supply side) energy management to increase the overall energy efficiency in the district. Multi-objective optimisation algorithms, again, can be applied to districts, as in the case of buildings discussed previously. They more or less manage the supply side of the districts, which deals with the operation of energy sources. Previously, a lot of the work focused on single-objective optimisation, aiming to only minimise the overall cost of energy generation (Buoro et al. 2014; Dorfner and Hamacher 2014; Cai et al. 2009; Murai et al. 1999). The underpinning research here addresses individual systems or technologies in the domain of district energy optimisation (for example, boilers, CHP, district heating networks, biomass energy, etc.). However, research from Pantaleo et al. (2014a and 2014b) details exceptions, where the aim is to optimise operations in a smart grid or microgrid; however, they focus solely on electricity or power domain. These studies lack a more holistic approach to systematically consider the inter-relationship among the different domains. Taking a holistic approach today is important especially because of the rising carbon emissions rate.

For example, increasing concern about climate change and $CO_2$ emissions has led to more stringent environmental legislations and hence, energy managers have also had to keep greenhouse gas emissions as low as possible. There is also constant pressure for improvement of the technologies and fuels used; as this helps keep emissions from energy production as low as possible. Integration of renewables and low-carbon energy sources into the generation mix has also become crucial as they help cut down on emissions level. For example, a lot of work has explored trying to reduce greenhouse gas emissions by integrating biomass plants into the generation mix (Pantaleo et al. (2014a and 2014b); Chicco and Mancarella 2008; Mancarella and Chicco 2008). These models were more or less focused on biomass supply chain and distribution; however, they do not look into operational optimisation. Biomass is a low-carbon option as it produces fewer emissions compared to other fossil fuels. Nevertheless, the costs of integrating renewables into the generation mix and its management need to be capped, prompting the need for a multi-objective approach.

Arnette and Zobel (2012) use mixed-integer linear programming (MILP) techniques to deal with these multi-objective problems, taking into account renewable energy and conventional sources of energy such as coal plants, but focus on the electricity supply side alone (Arnette and Zobel 2012). Their approach does not consider thermal energy.

Smart control of district energy management systems has been using neural networks, forecasting models, optimisation techniques in complex systems to support scheduling, adaptive control, model predictive control, and robust pattern detection. Some of these, however, focus on the design stages of a district. Hiremath et al. (2011) introduce a holistic mathematical model which takes into account several objectives, e.g. production and distribution price impact on the environment; efficiency of technology used; and also potential labour employment in the area due to decentralised energy planning. The model, however, is applied for design optimisation rather than operational optimisation. Research previously has also looked into optimising the control parameters in the energy systems – boiler setpoints, water flow set points, and district heating supply (Jamot and Olsson 201; Jiang et al. 2014). These set points or schedule of setpoints remain constant throughout the operations phase and do not take into account the day-to-day changes in demand profiles or weather, for example. A ‘master-slave’ optimisation technique (Fazlollahi and Marechal 2013; Fazlollahi et al. 2014) was used to combine evolutionary algorithms and MILP to solve the district optimisation problem. However, the research looks into both sizings of systems and operating parameters, which are more relevant for design purposes.

Similar to the above, Maifredi et. al. (2000) use a decomposition approach to solve their optimisation problem. This work, however, can be applied at the operational level; the dynamic programming theory is used to provide dynamic schedules (changes every 24 hours) for electricity and heat production in the co-generation system. The authors split the optimisation problem into a dynamic problem and a static problem with each having their respective set of decision variables, but the optimisation only considers cost as the objective.

Baños et. al (2011) conducted a review of optimisation methods applied to renewable and sustainable energy, and showed a significant increase in research papers using optimisation methods to solve renewable energy problems, especially for wind and solar systems. The authors’ review focused on papers that use traditional optimisation methods such as mixed-integer and interval-linear programming; quadratic programming and Lagrangian relaxation. They argue that heuristic optimisation, such as genetic algorithms and particle swarm optimisation, is a growing trend in the field of renewable and sustainable energy management. The review reveals that most of the research has not yet taken a holistic optimisation approach as the focus is more on individual renewable energy sources, e.g. wind power, solar energy, hydropower, and bioenergy. Their review also indicates that forecasting techniques are combined with optimisation approaches. For example, Marik et al. (2008) combine forecasting with mixed-integer nonlinear programming (MINLP) optimisation techniques, whereas Hashemi (2009) developed an offline model to optimise the operations of a combined cooling and heating power (CCHP) system (with storage) and uses non-linear solvers (LINGO 8.0). However, both studies only consider cost optimisation. Ikonen et al. (2014), used physical models and multi-integer programming to optimise the supply temperature of a district heating network; they propose to extend their work in the future using forecasting models to implement near real-time optimisation. Pini Prato et al. (2012), examine thermo-economic optimisation of CHP systems using MILP techniques. Even though this optimisation focuses on the operational stages, it does not consider emissions as an objective, but rather tries to exploit the heat storage capabilities of the network itself.

Summarising the review of the district domain, again, combinations of AI techniques have been the recent trend, similar to the building domain. Very few, however, have looked into multi-objective optimisation during operational stages which takes into account all the different domains – costs, emissions, and efficiency. Figure 9 below summarises the various optimisation methodologies adopted in the work covering this domain and shows that evolutionary algorithms are seldom applied, despite being capable of solving multi-objective complex problems, as seen in the building domain. Therefore, a future potential strategy is to apply evolutionary algorithms such as the GA to the issue of supply side management in districts.

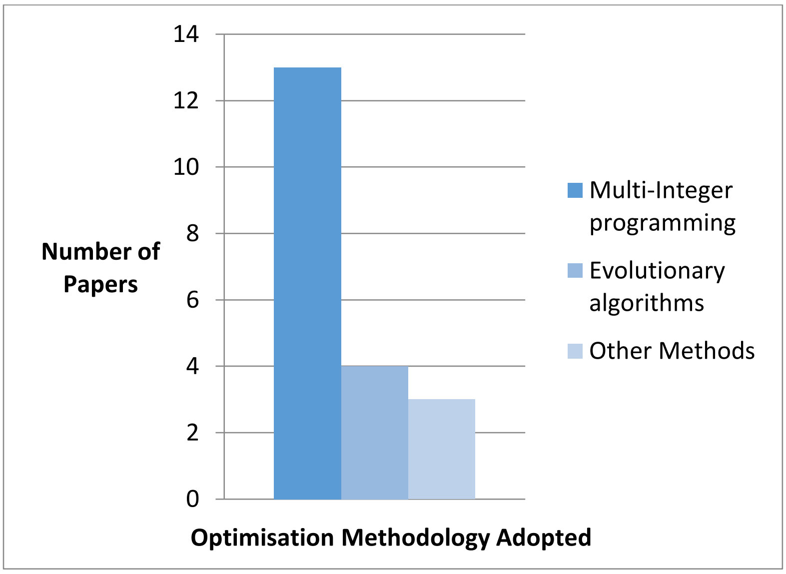



**Figure 9. Optimisation methodology adopted**

To apply such a multi-objective optimisation algorithm, however, a simulation or mathematical model would also be needed which could simulate the working of the district to compute the costs, emissions and efficiency there.

Optimisation models are complemented with multi-agent systems (MASs) for real-time energy management in districts. MAS have been used previously for operational planning in district energy systems (including district heating systems) and are becoming a trend

(Wernstedt et al. 2007). Doing this, however, also requires simulation models to simulate the dynamics of the district based on the decisions made by the multi-agent systems. Consequently, analysing the results of the simulation helps researchers study the effectiveness of the operational planning decisions (Johansson & Wernstedt, 2005). For example, Wernstedt and Johansson (2008) suggest that this is usually carried out using weather forecasts as an input to simulation models to predict the future heat demand for any period. Consequently, multi-agent systems can then take action to look into optimisation of the heat demand. Simulation models used here usually do not explicitly model the distribution network or the individual consumer behaviour. They tend to be embedded as a black box within the consumer node.

However, MAS are predominantly applied for demand response and matching of demand and supply in the electricity domain (Li and Nair 2015; Brazier et al. 2015). They are also used for distributed energy resources' (DERs) management for better coordination of supply and demand, again with a focus on the power sector (electricity domain). Demand response here means customers can respond to the changing supply conditions (for example, the market prices) and change the pattern of their consumption. However, this does not reduce the demand itself, which is important. Actual reduction of demand would require agents to work with numerical optimisation.

#### 2.1.4. Discussion

Evolutionary optimisation algorithms used for energy management in buildings and districts are certainly capable of increasing energy efficiency in the domain. These optimisation algorithms (evolutionary algorithms) need to be combined with mathematical/simulation models or other AI techniques such as prediction for better energy management. For example, using multi-objective optimisation with prediction models seems to be the way forward for real-time energy management and optimisation in buildings.

The problem, however, is that demand side optimisation in buildings does not work together with the optimisation of the supply side in their districts. Vice-versa, supply side optimisation does not take into account demand optimisation (i.e. reduction); it largely focusses on management of supply to meet the particular demand. Therefore, one of the biggest challenges today is to harmonise energy management solutions at building (demand side) and their district (supply side) levels.

Despite the advancements in research, there are couple of common challenges both domains face:

1. Both the domains need to look into all objectives possible for a holistic approach to energy management. Some of these objectives are listed in Table 1:

**Table 1. Multi-objectives to be considered for demand- and supply- side.**

| | Demand Side | Supply Side |
|---|---|---|
| Minimise Costs | ✓ | ✓ |
| Minimise Emissions | ✓ | ✓ |
| Minimise Energy | ✓ | |
| Maximise Comfort | ✓ | |

2. Solutions need to be linked with ICT technologies and automation systems to be implemented in real time and reduce the performance gap.

## 2.2. Ontologies applied today

This section of the chapter looks to review the application of ontologies in the domain of energy management in buildings and districts. The Scopus database was used to search for literature relating to the domain. The keywords used for searching were – 'ontology AND building energy', 'ontology AND district energy', and 'ontology AND 'energy management'. Keywords 'building energy' and 'district energy' were separated from the search as only five documents were found when the search combined these keywords with ontology. The operator AND was used with each of the keywords (building energy, district energy and energy management) because the scope of the review was limited to applications that use ontologies. The search criterion was also confined to the field of physical sciences, which includes some of the major domains such as Engineering, Computer Science, Energy, Mathematics, and Environmental Science. The search entered into Scopus is shown below:

*(TITLE-ABS-KEY (ontology AND district energy) OR TITLE-ABS-KEY (ontology AND building energy) OR TITLE-ABS-KEY (ontology AND energy management)) AND SUBJAREA (mult OR ceng OR CHEM OR comp OR eart OR ener OR engi OR envi OR mate OR math OR phys)*

The search listed 369 documents as part of the results. These documents were then further filtered by removing unwanted subjects such as chemical sciences, business-related work, management-related work, and social science-based work. By doing this, the number of documents was filtered down to 349. The abstracts of these papers were read and irrelevant literature was removed. Then, documents that met at least one of the following criteria were selected for the final review:

1. Linked to buildings domain.
2. Linked to districts domain.
3. Linked to energy management.
4. Ontologies used as a middleware in software design.
5. Ontologies used for decision-making.
6. Ontologies focused on energy systems as silos, such as PV systems, wind turbine, etc.

A total of 90 remaining documents were included in the review section. Apart from the literature taken from the Scopus database, the authors also reviewed some of the European research projects linked to the theme. The review helped categorise the work based on the methodology or level of application of ontologies. It was split into different categories, as shown in Figure 10 below:

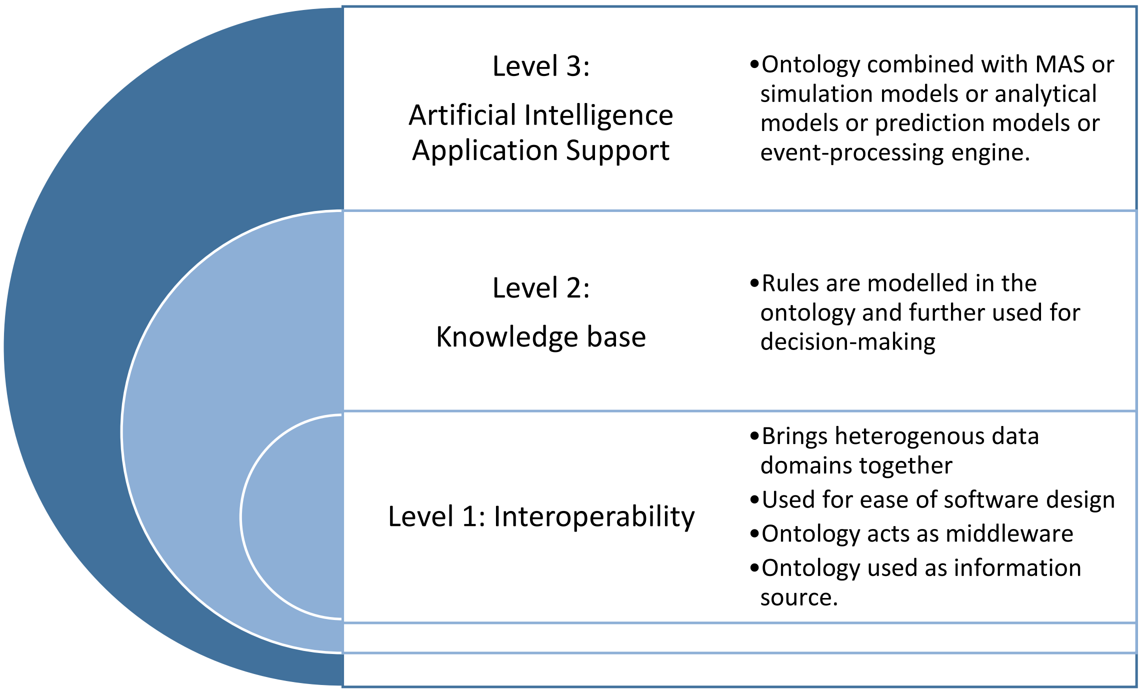



**Figure 10. Ontology application in the domain of energy management in buildings and districts**

- **Level 1: Interoperability** – this is the very basic use of an ontology when it is used merely to map different domains or bring heterogeneous technologies together. It can be further used to build independent domain applications. The application can query the ontology model to retrieve any static information. In the case of dynamic information, the ontology can be used to point to external databases or systems that store the location of these dynamic variables. For example, it can store the parameter location of variables from automation servers (which operates the automation systems), through which the current value can be read. Reasoning, as usual, can be applied to these ontologies, which may infer new knowledge that was not specified in the ontology explicitly.
- **Level 2: Knowledge-base** – on top of interoperability, ontologies can also store rule axioms (SWRL rules or any other rule language) within the model. The rules make sure that certain conditions are followed if certain other conditions happen. These rules can add intelligence to the model; for example, energy anomalies or wastage situations can be identified in a building domain ontology as long as these conditions are pre-defined as rules in the ontology. They also help the reasoning purpose.
- **Level 3: Artificial Intelligence applications supported** – here the ontology can be further linked with simulation models, multi-agent systems, event processing engines, etc. The ontology supports the running of these applications by providing the necessary information to the end-users for decision-making.

The sections below review some of the research previously held in each of these categories in detail.

### 2.2.1. Ontologies Used for Interoperability

Here, the ontologies are solely used to bring different data domains or technologies together.

Many works use ontologies to represent home appliances and devices. **DEHEMS** (Digital Environment Home Energy Management Systems) ontology includes knowledge on different home appliances (Shah et al. 2011; Shah et al. 2010). Therefore, ontologies facilitate interaction between two or more heterogeneous applications. The **DEHEMS** ontology here is linked with Suggested Upper Merged Ontology (**SUMO**), which acts as an abstract layer on top of the domain ontology, the largest high-level ontology today (Pease 2016). Reusing or linking existing ontologies with domain-specific ontologies is highly encouraged for ontology engineering to promote interoperability and reusability (Noy and McGuinness 2001). The SUMO framework is an upper-level domain-independent ontology which enables disparate systems to utilise common knowledge. From **SUMO**, other domain-specific ontologies can be derived and it facilitates metadata interoperability and knowledge sharing among other SUMO-compliant ontologies. Compliance with **SUMO** makes the domain ontology more generic and easily reusable (Cherifi 2011). Other works using **SUMO** ontologies are A. Sicilia et al. (2015), Shah et al. (2010) and Tomašević et al. (2015).

**SAREF** (Smart Appliances REFerence ontology) is another ontology used to represent smart appliances used in households. The ontology was developed to help develop a standardised interface for sensors and devices. The ontology can be used to communicate to heterogeneous systems found in the households (Daniele et al. 2015). Very similar work was also carried out by Den Hartog et al. (2015) to create a reference ontology for smart appliances in residential households. However, these ontologies were based on home appliances alone and lack supply and demand side concepts.

Wicaksono et al. (2010) used an ontology to integrate many heterogeneous technologies, which when combined can support an intelligent energy management system. Here, the proposed KEHL framework uses a knowledge-base approach based on ontologies to represent knowledge. The knowledge was generated in three ways: 1. Manually by domain experts, 2. Semi-automatically by interpreting 2D drawings, and 3. Semi-automatically using data mining algorithms. The ontology is also linked with other modules:

- A *presentation module* (for visualisation, configuration, and control of building automation systems);
- *2D drawing interpretation module* (for semi-automatic knowledge generation from 2D CAD drawings using JavaScript rules); and,
- *Data acquisition module* (collect data from different building automation logic control units or gateways. The data is stored in an SQL-database where it is pre-processed and ready to be used by the data-mining algorithm to derive energy usage patterns).

Some of the work covered here is focused on the building design stages. For example, Niknam and Karshenas (2015) made available disparate sources of data relevant to energy analysis through semantic web services. Information such as geometry, material properties, mechanical equipment specifications and climate information was used for energy analysis. The energy analysis applications could automatically discover and retrieve this data. The information was made available over the web in a machine-processable format which could be shared, accessed and combined through a semantic web. These services allowed the building designers to focus on building design optimisation rather than spending time on data preparation and manual entry into energy analysis software. Similarly, Katranuschkov et al. (2015) used ontology as a platform to bring together multiple information sources in multiple different data models in a consistent manner for the efficient functioning of the Virtual Energy Lab. The Virtual Energy Lab was built for holistic building energy analysis during design stages. Although this work was used during the design stages of buildings, it shows the true potential of having an ontology in this building energy domain, which is complex in nature due to the multi-domain information models involved.

Ontologies were also used as a middleware to enable the bringing together of various kinds of services to facilitate building energy management. Cafarrel et al. (2012, 2013) proposes an ontology-based multi-technology energy management platform called 'Bat-MP', which enables integration of various building automation protocols, linking various kind of services, and allowing sharing of data. Bat-MP is designed as a middleware to support different kinds of services capable of managing building systems through home automation protocols. The middleware consists of three layers – technology manager, model manager, and service manager. Out of these, the model manager consists of the ontology which is used for the description of characteristics, properties and entities of a building. The service manager layer can then provide an application programming interface allowing a service to connect to the platform and interact and access information related to parameters in the building. From an energy management point of view, the Bat-MP reads information from sensors and control actuators from different building management technologies. It can also implement demand response services by communicating with utilities to retrieve metering information. Hence, it can help reduce building energy consumption.

Meanwhile, Project IntUBE aims to improve the energy efficiency of buildings by developing intelligent ICT techniques (Böhms et al. 2010). According to the authors, "*the ontology used here brings together existing software functionalities using open standards and their open source implementations.*" It advises users on their energy-consuming behaviour to minimise consumption while maintaining comfort. Ontology is used here as an integration tool to bring together the various information needed for IntUBE to make decisions. It brings together four repositories:

- BIM (Building Information Model) repository – contains all the general static information about the building such as location, building services, spaces, etc.
- SIM (Simulation Information Model) repository – contains dynamic information which is needed as input parameters for simulation programs.
- PIM (Performance Information Model) repository – handles the monitoring of data that is dynamic and changes in real time;
- RD (reference data) repository – contains the metadata which stores interlinks between data from the above three repositories.

Authors such as König et al. (2011) and König and Stankovski (2012) complement work in project IntUBE by integrating all the relevant information from various sources, for the stakeholders along the building lifecycle. The data here includes information that is mainly linked to Energy Efficiency (EE) and Renewable Energy Storage (RES) aspects of the buildings. Integration was carried out by using an ontology to build a sustainable building profile. There is currently a general lack of approaches that integrate factual data and information from a variety of sources and hence this ontology was helpful here, being an open knowledge-base. Along these lines, Pauwels, Törmä et al. (2015) investigated the use of semantics to bring together different data domains that can also compliment the Building Information Modelling (BIM) process when it comes to managing energy information. BIM is a process in which stakeholders and companies from the Architecture Engineering and Construction (AEC) industry who are involved in a project share information throughout the lifecycle of the project. BIM is simply information modelling and information management in a team environment. Certain applications require more than just BIM information (National Building Specification 2016). For example, Costa and Madrazo (2015) through a linked data approach, added semantic information about building products from multiple sources to building components in a BIM model, thereby giving structural modelling more meaning. Combining BIM with additional data models can also benefit the building energy analysis process.

Currently, Industry Foundation Classes (IFC) data model, the standard for the BIM process, is not good enough to carry out holistic energy analysis. There is a gap in the IFC which does not give all the information needed to perform building simulations. Consequently, it fails to transform into a simulation domain model and cannot be used as a direct input for performance simulations (Thorade et al. 2015). Information models for the simulation domain and corresponding file formats have been developed to deal with these drawbacks. The multiple sets of information that ontologies bring together can be fed into simulation tools and related simulation models. For example, energy performance analysis requires information on climate, architecture of building, user behaviours, etc. An ontological approach is beneficial as it is capable of following a centralised integration with the BIM model as the centre. BIM model implementations like the IFC aim to provide general concepts to cover common building design scenarios.

Kadolsky et al. (2014) used ontologies to conceptualise the building, its external data (such as climate) and the relationship between these. Logical rules are used to represent the constraints and the calculation methods. These were then enabled through the energy-enhanced BIM (ee-BIM) framework, which was consequently used to pre-check the simulation input and pre-analyse the energy performance by applying inference rules. The framework has to offer non-BIM domain models to complete **ee-BIM** ontology. The ontology platform, manages the input data, also helps in the execution of calculation methods.

Especially for real-time energy management, looking into a single domain is not enough, and additional information from various interlinked domain needs to be taken into account (Kadolsky et al. 2015). Similarly, Corry et al. (2015) suggested ontologies can bring more than just BIM data to close the performance gap. Results provided by current BIM-based energy performance tools have been criticised, and they deviate from actual data measurements because of modelling assumptions and simplifications (Ham and Golparvar-Fard 2013). The building automation system (BAS) modelling needs to be interoperable with BIM (Scherer et al. 2012) so that real-time data can be taken into account, consequently closing the performance gap. The authors here propose using ontologies to bring these two worlds together. Energy-efficient design and operation requires data and information models that do not originally belong to the Architecture Engineering Construction (AEC) and Facility Management (FM) domain. Hence, these need to be linked to the BIM model. In similar works, Muthumanickam et al. (2014) investigate linking an IFC-based BIM model with energy-consumption data. Ploennigs et al. (2011) also developed an ontology **HESMOS** which closed the gap between information from BIM and monitoring data in building automation systems.

The Common Information Model (CIM) (UCA International Users Group 2014), one of the leading standards in the energy management domain, was used for development and alignment of the facility ontology. The IEC 61970 series of standards (CIM model) deals with the application program interfaces for energy management systems (EMSs). The Industry Foundation Classes (IFC) (buildingSMART International Ltd 2016) data model, which is a standardised specification of Building Information Modelling (BIM), was also used here which helped define the basic domain entities such as the data types, devices and (sub) systems. This work is also part of European energy research project CASCADE, which uses the ontology developed for facility energy management.

### 2.2.2. Ontologies used to develop knowledge-bases for decision-making

Kofler et al. (2011, 2012), developed an ontology to model an energy knowledge-base that provides information to users on energy consumption and also allows the home automation system to make intelligent decisions to optimise energy use. The authors above extended their work in 2013, and explored capturing user profile information and its effects on optimisation tasks in an ontology for future smart homes (Kofler et al. 2013). Representation of user behaviour is important to achieve greater comfort while trying to improve energy efficiency. The idea behind this work is to retrieve the already existing information from the BIM and AEC domain, and consequently integrate this to aid decision-making in smart homes. The objectives of such a system are to maintain user comfort and increase energy efficiency (Kofler and Kastner 2010). The ontology developed can be used as a shared vocabulary for an agent-based software system. The ontology also considers renewable energy suppliers which help the residents reduce their ecological footprint. It is formed of various modules which consist of different categories of parameters (useful for energy efficiency accordingly). Figure 11 below shows some of the main modules of the smart home ontology.

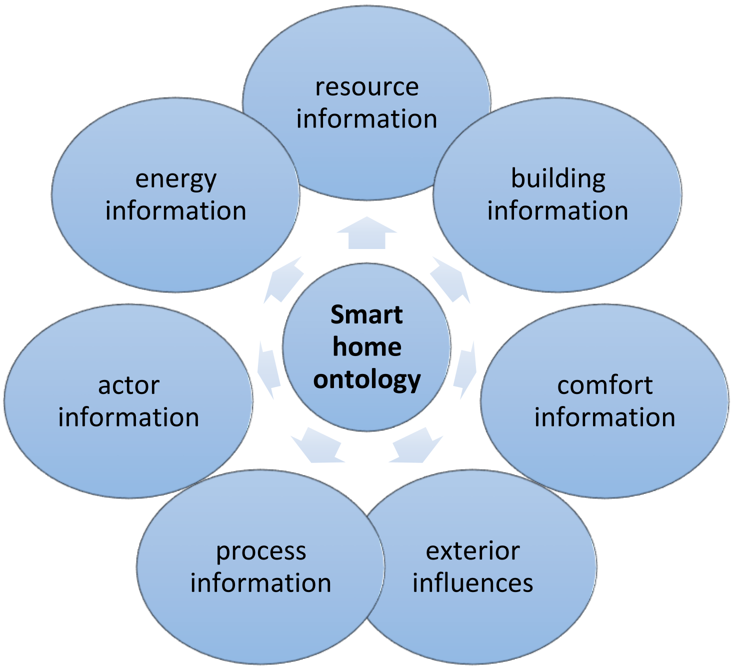



**Figure 11. Different modules connected to smart home ontology**

The ontology here has a special focus on both the demand side and supply sides, and the interrelationships between them. Some of the applications of this would be to make sure the household appliances and consumer electronics run efficiently; and taking decisions on which renewable energy supplier to use would provide the optimal tariff for a scheduled task or program. This global system for intelligent, smart home management is called ThinkHome (Reinisch et al. 2010).

Ontologies can also help derive SWRL (Semantic Web Rule Language) rules which can regulate system behaviour at a home automation level. Valiente-Rocha and Lozano-Tello (2010) created an ontology-based expert system which helped home automation and rules management. The model contained a database of home devices where the attribute values could be stored and also used to instantiate the ontology. A software application called DomoRules was consequently used to create SWRL rules from these instances. IntelliDomo, a system for controlling a domestic automation system (domotic), was used to draw inferences from the ontology and SWRL rules, by using parameters that were last indicated by the user, which meant that the state-of-the-art devices could be modified in real time. A very similar kind of approach was taken by Wicaksono et al. (2015), where the ontology was developed for intelligent energy management in buildings during the operational stage. It uses ontologies as a knowledge-base to represent the intelligence and then further uses it for reasoning purposes. Here, the knowledge in the ontology was represented as rules and was used to identify energy wastage situations. The rules can be derived manually by domain experts, or semi-automatically using data-mining algorithms (provided monitored results are available), as shown in this study (Wicaksono et al. 2010). These rules can then be used to draw conclusions on building performance. The work was extended in 2013, where the authors used a rule engine based on SPARQL_Jess Bridge, which combines the execution of rules with Protégé API (Wicaksono et al. 2013). ifcOWL ontology was also included in this work.

**`PowerOnt`**, which is an ontology-based power consumption model for smart homes, was developed, where the authors used it to estimate the real-time power consumption in homes with very few meters (Bonino et al. 2015). Knowing the power consumption level can encourage householders to make positive changes to their energy usage behaviour. This ontology is also capable of linking with other smart home environment ontologies as the research links **`PowerOnt`** with **`DogOnt`** ontology. Nuccci et al. (2013) developed an ontology for a smart home, where all elements of the smart home (services, context, and devices) are formally described but with a focus on device and energy ontology. The novel holistic approach presented here can deal with energy production and consumption along with device and services management. Rosello-Busquet at al. (2011) also created an ontology to bring together the various electrical devices and appliances in a household to optimise the energy use and increase energy efficiency. The ontology here deals with the issues of interoperability when it comes to bringing devices from different vendors together. This ontology was used in the home energy management system (HEMS) proposed in this work. The HEMS also contains semantic web-based tools that provide a common interface and can also suggest energy management strategies through SWRL rule implementation (Rossello-Busquet et al. 2011). Ontologies used by Sallinen et al. (2012) provided data from different sources to an intelligent service platform, which further made decisions and controlled appliances in homes based on the information. By using ontology mapping, the platform was able to exchange information between different applications. The JESS engine was used to make decisions, and the platform as a whole was implemented using the OSGi service framework to which other systems could easily connect. One of the applications demonstrated through the platform was smart home heating.

Grassi et al. (2011) talk about how energy consumption and production need to be dealt with together, and consider a holistic vision for the smart home environment. The ontology framework discussed here provides the necessary information needed for its implementation. However, the work is only focused within the smart home environment – dealing with energy generation within the home environment. Semantic web technologies were used to provide data interoperability between the various devices involved and also to provide inference capabilities for task management and decision-making, aiming for higher energy saving.

Ontologies have also been used for office activity recognition (Nguyen et al. 2014). Activity recognition can be useful as it is a key input for building energy and comfort management systems. Knowing the activity can help control appliances to save energy while still meeting the comfort requirements. Again, a rule-based reasoning feature was used to identify the activity based on the state of all sensors. Similar work was conducted by Georgievski et al. (2013), where an ontology-based activity recognition was combined with AI planning for control of appliances. Camacho et al. (2014) used ontologies for conflict detection in home automation systems, thereby improving energy efficiency and maintaining comfort. Hsu and Wang (2008) used ontologies for smart home resource management where energy consumption, living space partition and network bandwidth (to detect any resource conflict) are all modelled into the ontology. Consequently, Inhabitants' resource requirements can be predicted using the case-based reasoning features of this ontology. Ahmadi-Karvigh et al. (2016) used ontologies for pattern recognition and detected unsustainable behaviour leading to wastage of electricity. Such a study further encourages occupants to change their behaviours. The appliance data usage is input into the system proposed by authors. This input data is then categorised using ontologies based on their context information. The data is further segmented into active and inactive segments. Furthermore, the active segments associated with electricity consumption are then estimated. Hong et al. (2015) developed an ontology to represent energy-related behaviour of occupants in buildings. According to this study, the occupants' interaction with the building can impact the overall building performance including comfort, energy load, technology efficiency and operational costs. Therefore, it is important to account for these behaviour changes and their effects on the building performance. The framework used **DNAS** (Drivers Needs Actions Systems) ontology which has four parts:

1) Drivers of behaviour (this is dependent on the environmental conditions that prompt actions from occupants to fulfil their psychological or physical needs),

2) Needs of occupants (these are the tangible and intangible requirements of occupants for environmental satisfaction),

3) Actions by occupants (these are the actions that occupants perform to interact with systems),

4) The systems used by occupants (these are the systems within or outside the buildings with which occupants can interact to control their indoor environment).

Building modellers use this framework during design stages to simulate occupant behaviour, and, during the operational stages, the predictive models and algorithms can help improve energy performance by advising users through smart technologies. Engineers can also find this tool helpful as they can review their technologies or system's performance by simulating the impact of different energy-related occupant behaviours. Such a review can be useful both during design stages and for retrofits.

The SESAME project (Sesame Project Consortium 2011; Tomic et al. 2010) was focused on using smart meters and sensor-enabled solutions for buildings, as well as adapting them for commercialising in real-life settings. The project used semantic web technologies and semantically linked data to help users control their energy usage by making informed decisions. A number of ontologies were used – SESAME Automation ontology, SESAME Meter Data Ontology, and SESAME Pricing ontology. Rule-based policies were also used here to provide decision-making.

Ontologies help design and manage low carbon/energy buildings. The authors in Tomašević et al. (2015) implemented a facility data model using ontologies as a part of the contemporary semantic web paradigm where the ontology stores the static knowledge. The use cases presented in the study show how this static knowledge was used for decision-making and how it helped improve energy management by interfacing this with a custom-based API (Application Programming Interface). The project investigates reducing the energy needs of airports by developing an ISO 50001 energy management system. The system helps with advanced fault detection and can be used as a diagnosis tool. The ICT solutions developed can also integrate with building automation and management systems for reduction of energy consumption by 20% and $CO_2$ emissions by 20% (PSE AG 2012). This ontology was used in the European energy research project CASCADE, mentioned previously, for facility energy management. Tanasiev et al. (2015) also used ontologies for improving energy efficiency and maintaining user comforts. Kim et al. (2014) used an ontology-based knowledge-base cloud service for sharing knowledge with various BEMS. Once integrated with the ontology, BEMS can access the knowledge required for improving energy efficiency in buildings. Such a solution integrates the context information (intelligence) gained from various BEMS experts and provides context-sensitive rules for a non-domain expert, thereby sharing knowledge on energy management. The knowledge cloud connects the building information, facility information, energy information, and environmental information needed for holistic energy management decision-making via the internet. This knowledge can accumulate over time and users of the service can benefit from each other's knowledge. Yuce et al. (2015) used ontologies to store rules (in SWRL format); consequently, the ontology could be queried by the facilities manager through a unique user interface that allows negotiation of energy-saving measures to occur. The rule-generation methodology in this paper is unique as it used simulation models, ANN models, and optimisation algorithm; and this system was used in the EU FP7 project KnoholEM (Howell et al. 2014; Yuce and Rezgui 2015). The end product of this was an intelligent building energy management system for public buildings through a holistic knowledge base, as described in Anzaldi et al. (2012). It provides an energy management strategy by focusing on occupants' behaviour in buildings. The platforms bring together building occupants' activities, the interaction between facilities and users, and building infrastructure information to improve building energy usage (Sicilia et al. 2015). The ontology of the repener-linked dataset was focused on the building energy performance domain containing information from the entire building life cycle which affects energy performance. The ontology also includes elements from different standards and covers certain core areas such as general project data (location of the building, use of the building, etc.), building properties, weather information, and operational data (thermostat setpoints, occupancy, comfort parameters), and certification (energy ratings). The dataset makes the end-users' decision-making process easier as they have access to multiple sources of information, and this can improve the energy efficiency of buildings. Hou et al. (2014, 2015) used semantic web technologies to help structural engineers take a sustainable approach while designing. The ontology recommends structural design solutions that have low embodied energy and carbon as this is linked to the knowledge base, which combines structural information, materials and their environmental data.

Ontologies have been used to aid energy management systems that apply to both domestic and industrial fields. Lopez et al. (2015, p.168) developed the ENERsip ontology which, according to the authors, helps *"formally define the vocabulary and taxonomy and captures the engineering and business semantics of domain of knowledge of energy efficiency platforms needed for nZEN (nearly zero energy neighbourhood) vision of smart* *grids".* Nearly Zero Energy Buildings (nZEB) are simply buildings that have a very high energy performance and consume nearly zero or very low amounts of energy. They can use energy from renewables, which can either be produced on-site or in the vicinity (European Union 2010). The European Performance of Buildings Directive (EPBD) states that, by 2020 all new buildings needs to be nZEB, and this also should be extended to existing buildings undergoing retrofits (Marique et al. 2013). Nearly Zero Energy Neighbourhood (nZEN) is the extrapolation of the nZEB concept wherein annual energy consumption of buildings in a neighbourhood and transportation of its inhabitants are balanced by renewable energy production locally. Macek et al. (2011) developed a platform, 'ENERsip', for industrial and residential users in households or flats, aiming to provide information about their electricity consumption, and consequently guiding them to make energy-saving decisions. The ontology here makes the implementation of services easier and significantly makes system development faster. It is primarily used here to help the software design process and not to make decisions on energy savings.

Monitoring of real-time data and simulating the energy flows in a district is needed for optimisation of energy consumption. There can be many sources of such information, and it can also come encoded in different formats. DIMcloud is a model that helps the integration of heterogeneous data at a district level using ontologies and also establishes the relationships between this data. Ontologies play a significant role in DIMcloud as it acts as the link to gathering all the information required from the various types of databases (Brundu et al. 2015). This however lack any building level concepts and do not look into demand side optimisation. In similar work, Q. Zhou et al. (2012) used semantic web techniques to create an integrated smart grid information model. The model here is used for integration of information and knowledge representation which will help towards next-generation smart grid applications with a focus on demand response. An event-processing engine is linked to the ontology for decision-making. In summary, the semantic model captures the following (Zhou, Simmhan, et al. 2012):

- Data Sources: the sources of data can be smart meters (measures power consumption), sensors (airflow, occupancy, and temperature sensors), and weather-reporting services. The physical and virtual spaces that the data is measuring are linked to these sources.
- Infrastructure: the distribution network and the physical environment of the campus power grid infrastructure are modelled as well. Buildings, rooms, energy appliance concepts, and their relationships are included here.

- Organisation: the organisations in the campus are also modelled here, which includes departments, labs, schools, etc. These can help define demand response strategies; for example, a department coordinator can be alerted in the case of consumption that exceeds the threshold.
- Other information: existing domain ontologies are also used which can help model information such as scheduling and weather.

Authors in Kuriyan (2015) developed a software framework which is used to analyse urban energy models. One of the parts of the framework is a technology database which is implemented as an ontology developed in Protégé. The ontology here is used simply to describe the available energy conversion, storage, and transportation processes. The ontology then feeds information to the framework for it to analyse scenarios for design optimisation of the urban energy systems while meeting objectives and constraints such as investment costs and emission targets.

### 2.2.3. Ontologies used to support AI techniques

Ontologies can also be used in combination with simulation models. SimModel (simulation domain model) is an interoperable XML (Extensible Markup Language) based data model used for the building simulation domain (O'Donnell et al. 2011). Pauwels et al. (2015) developed a SimModel as RDF graphs to make it interoperable with the other building information and data. SEMERGY is a computational environment which helps in building design and refurbishment optimisation by using semantic web technologies and simulation models (Gudnason and Scherer 2012). The framework embeds a comprehensive building data model, and it uses an ontology for the building product data. To consider valid building construction alternatives, SEMERGY deploys a rule-based logic. The refurbishment optimisation considers investment costs, environmental impact and energy demand of the various alternatives (Wolosiuk et al. 2014). The authors in Han et al. (2011) used four ontologies in their system architecture to provide an intelligent energy management system. The 'building architecture' ontology contains general information on the building's areas and zones. The 'context' ontology is used to identify if the building is behaving abnormally by analysing the operational data. The 'cause' ontology points out the cause of the abnormality. Finally, the 'control' ontology implements actions to solve the reasons for this abnormality. Abnormal situations lead to energy wastage. In addition, the simulation model was also used, the results of which helped accumulate inference rules by studying the effect of varying key control parameters on building behaviour. The study, however, does not state if these rules are generated automatically or manually.

An ontology-controlled energy simulation was developed by Baumgärtel et al. (2015). Here, an enhanced version of **`ee-BIM`** is used to support the integration and validation of input data for the simulation. This is achieved with the help of ontology constraints and rules. The work is beneficial because it automates the process of running various simulations to study different green building design options until the user-defined targets are fulfilled. It is also supported by parallel thermal simulations to run simulations of the various design choices.

Ontology was also used in combination with MAS and IEC 61499 function blocks to make BAS more intelligent for efficient energy management (Mousavi and Vyatkin 2015). BAS are heterogeneous systems that can have control over various devices (from the different vendors). However, these devices or their operating standards might be incompatible with each other. Ontologies ensure interoperability between the devices and their operations. The collaboration of these devices can help towards holistic energy management, making it more intelligent than a silo-based approach to energy management. The authors mention that the collaboration requires sharing of knowledge in a format that is machine-readable and machine-processable. For building automation systems this can be about facilities, devices, operations, building energy, etc. Once this is represented in the ontology in a machine-readable format, it can be queried, and used as a knowledge-base for decision-making. Ontologies represent knowledge in a structured way which deals with the increasing complexity of this domain, which can have many stakeholders, economic and environmental challenges. Once the ontologies conceptualise the domain, it can be reused for similar applications on different sites or systems. Moreover, the reasoning techniques that an ontology offers can be used for case-based reasoning, which is a powerful method for problem-solving.

Ontologies have also been applied at a larger scale. A city-level application of ontologies for energy management was demonstrated in a European Union Framework Programme 7 (EU FP7) project called SEMANCO. It aimed to develop an ontology-based energy information system and tools to help stakeholders involved in urban planning make decisions to reduce $CO_2$ emissions at a city level (Madrazo et al. 2012). Nemirovski et al. (2013) describe the method in which ontologies can be designed to make this possible. The ontology brings together various data sources from different scales and domains (building, urban), and consequently simulation and other assessment tools developed within the SEMANCO framework interact with the semantically modelled data (Corrado et al. 2015). Some of the domains brought together through the SEMANCO ontology are building, geospatial, energy, climate and socioeconomic (Sicilia et al. 2015). The work presented here integrates the semantic model developed with existing simulation software such as URSOS[2].

The Optimus project[3] aims to manage energy production and consumption but within a building level. It is an ongoing project which seeks to develop a semantic-based decision-support system to optimise the energy use in public buildings. The tools developed in the SEMANCO project, described previously, are further being used here to develop the decision support system optimises energy by providing short-term decisions. The decision support system analyses five types of data from heterogeneous and dynamic sources: building energy performance, social behaviour, weather forecasts, renewable energy production and energy prices (Á. Sicilia et al. 2015).

The semantic framework here feeds the decision support system engine with all the information needed. The engine further uses intelligent rules to propose action plans for the user. The intelligent rules are fed by predicted data, static data, and real-time data.

Ontology in Schiendorfer et al. (2015) was used to model resources from multiple hierarchical levels in a smart grid. The unified model is then used along with constraint-based optimisation algorithms to help distributed energy management. Fernandez et al. (2013) used ontologies along with multi-agent systems for management and control of a smart grid. The paper discusses the architecture where ontologies form the middle layer, which is the link between real-time data collected from entities in the smart grid and multi-agent systems used for decision-making.

Similarly, Aung et al. (2010) used ontologies for real-time operation and control of a smart grid. The ontology is used in the structure of the messages, and the agents interact with each other through these messages, providing a shared understanding of the information between the agents which work together for microgrid operation. The Ambassador project aimed to develop and experiment systems and tools that optimise energy usage at a district level by the management of energy flows, prediction and tracking of energy consumption and production. Doing this would meet its overall objective, which is to reduce the cost of energy in a district (Amires and

[2] http://ursos-software.com/
[3] http://www.optimus-smartcity.eu/

Bumblebeestudio.eu 2016).The project plans to investigate energy efficiency at both the district and building levels. It proposes to use an ontology for the autonomous system that needs to be developed. These autonomous systems aim to take real-time energy optimisation strategies to meet the various objectives. Here, the district energy management and information system (DEMIS) is designed to interact with other smart systems (BMS, EMS, etc.) to gather information which will help solve problems at the district level. Resilient (Resilient Project Consortium 2012), another project under the FP7 framework, also proposes the use of a district information model based on an ontology combined with simulation models and multi-agents to make optimum decisions at a district level. The aim is to help minimise energy use and consequently cut down carbon emissions (Hippolyte et al. 2014).

The Inertia project (Inertia Consortium 2012) looked into developing a data management infrastructure that allows the production and consumption of electricity to be measured, reported and controlled. The demand side management framework enables local and global multi-agent management at a building and grid level respectively. The project develops a building-level ontology that covers three domains: location, devices, and occupants. The ontology also describes the DERs (such as HVAC and lighting), the sensors, and the actuators installed. The ontology here refers to the energy-related BIM model and is used to bring together the DERs with sensors and actuators. This is used to meet the various needs of the project, especially for information gathering.

### 2.2.4. Discussion

Summarising the review conducted above in terms of the application domain: ontologies have been applied to three types of domain within the supply and demand sides– Smart Homes, Smart Buildings, and Smart Grids. There are also occasional overlaps wherein some of these semantic models cover aspects of another domain as well, but largely they can be split into three. Figure 12 below shows an overview of the application of ontologies in the field of smart energy management for demand and supply sides. It also highlights the various gaps in the field.

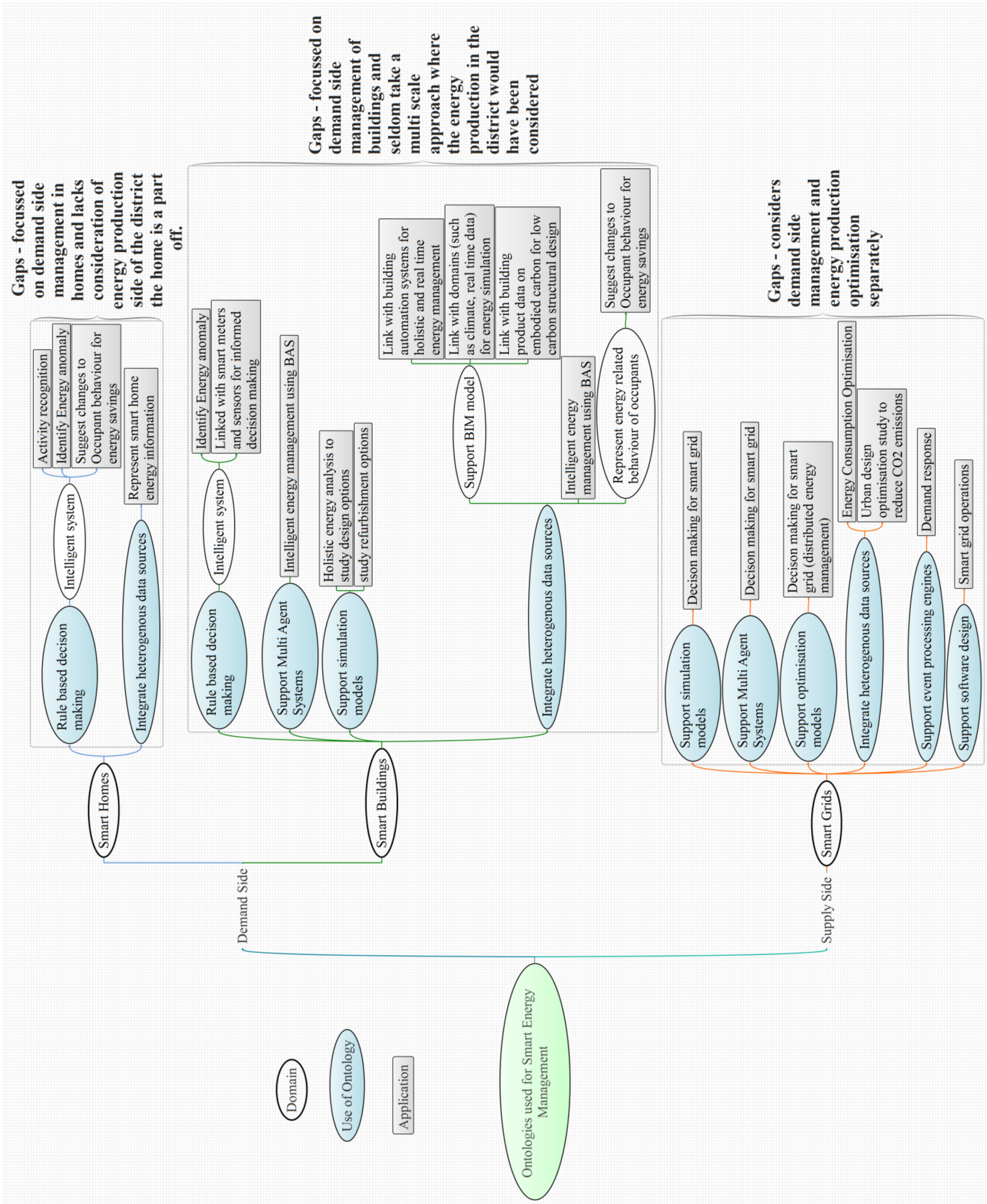



**Figure 12. Literature review summary with gaps in the field.**

The gaps, shown in figure 12, in each of the domain is summarised below:

*Ontologies applied in Smart Homes* - One of the major application areas for ontologies is smart home environments (SHEs). Integration between the various devices, appliances, and technologies is key for a holistic, intelligent energy management approach. Ontologies at a home level have mainly been used to represent the domain as concepts and also represent the interrelationships between these concepts. In other words, they have focused on presenting the information collected (data originating from devices, appliances, energy information, tenants, etc.). However, representation of energy information alone is not enough. Hence, following from this, the SHE was combined with reasoning and knowledge (modelled as rules), enabling it to be an intelligent system. The systems or frameworks used were capable of defining a common way to communicate with various devices from different vendors, for example, by integrating the various automation protocols. Most importantly, by doing this, they were able to contribute to making intelligent decisions. Some looked into activity recognition of occupants and intelligently controlled the appliances/devices in homes. Others monitor devices and occupant behaviour and suggest ways to positively change occupant behaviour in detecting energy-wasting situations. Positive changes to occupant behaviour were also made possible by using ontologies to estimate power consumption and analyse this. Most of these applications aim to minimise energy consumption and maintain occupant comfort. However, it is important to consider not just the demand side in these homes but also the supply or production side for holistic energy management, but research has seldom addressed integrating supply and demand concepts in a smart home.

*Ontologies applied in Smart Buildings* - The work reviewed under the building domain examines building operations in a smart way, aiming to achieve high energy efficiency and maintain comfort for users. For example, SWRL rules were modelled into ontologies to help detect energy-wastage situations. Some of the work also looked at the design stages, trying to achieve a low carbon design and also perform a holistic energy analysis in these buildings. Ontologies in the building domain were combined with simulation models for a holistic energy analysis to study various design options. They were also used to examine refurbishment options considering investment cost, environmental impact and energy demand of the different scenarios. Multi-agent systems were also combined with ontologies for intelligent energy management. Ontologies linked smart meters and sensors together to help occupants make informed decisions about saving energy in buildings. The review also suggests how ontologies can complement BIM information by combining various data sources with BIM data. Such an approach can help achieve a holistic energy analysis – as IFCs (BIM data standard) alone are not currently fit to do this. More importantly, to ensure holistic and real-time energy management, both static knowledge and operational knowledge need to complement each other. In relation to this, some research has tried to use ontologies to bring together the building automation systems and BIM model. Doing this is important, as it can lead to a reduction in performance gap and today work rarely addresses the integration of BEMS or BAS with this operational knowledge to aid in energy management. The problem here is also similar to the smart home environment domain – it is necessary to consider a holistic viewpoint and involve the supply side as well for decision-making.

*Ontologies applied in Smart Grids* - Another domain of application of ontologies is smart grids or in urban energy systems. In most cases, the ontology is used as a knowledge base for information brought together from heterogeneous sources. Some researchers used MAS or simulation models with a knowledge base to make decisions on smart grid operations. In some cases, ontologies have been linked to an event-processing engine for operations such as demand response. In certain cases, the ontology was used to aid software design as well, such as Macek et al. (2011), where the ENERsip ontology was used to produce semantics to achieve the nearly zero energy neighbourhood (nZEN) vision of smart grids. Ontologies are used for the design of autonomous systems, which aims at operating the smart grid in real time. They were also used for an urban energy design optimisation study where the knowledge base provides all the heterogeneous information (from various domains in a district or city) needed to assess urban energy systems, especially their design, from a cost or sustainability point of view. However, there is a lack of holistic approach because optimisation of demand within the buildings and production side optimisation are not considered simultaneously. The solutions reviewed here focus on the operations of the production side of the grid where it tries to deal with how best to meet the demand and never attempts to reduce the demand itself.

# 3. Ontology Development Methodology

From Chapter 2, it is clear that semantic models are applied for smart energy management in both demand (smart homes and smart buildings) and supply side (smart grids). Within their respective domains, both multi-objectives (user comfort, energy, environment, efficiency, and cost) and real-time data are considered for operational stage energy management. However, the integration of energy management techniques at demand and supply side is a definite research gap. For example, in previous works, smart grid ontologies represent the energy demand for buildings but do not implement or support techniques to optimise it. On the other hand, smart home/building semantic models look to increase energy efficiency within the home or building but fail to consider energy efficiency at their supply side. This PhD research develops an ontology to support holistic energy management for both supply and demand sides. The ontology to be developed was named **REMO**, which stands for real-time energy management and optimisation ontology.

The review also shows how semantic models linking with automation systems are increasingly used for rule-based decision-making to increase energy efficiency at a home or building level. On the other hand, at the district level, semantic models are used to support AI linked simulation models to increase energy efficiency. This research, along similar lines, focuses on developing an ontology-based framework which supports both AI models and automation systems to increase overall energy efficiency.

## 3.1. Ontology development and validation methodology

There are many ontology development methodologies that have been adopted in the past. Some of these works have been reviewed by Fernández-López (1999). Looking at some of these methods presented in the paper above, a few of the common steps involve:

1. Definition of the requirements (where the purpose of the ontology is defined).
2. Extracting terms and concepts (here the taxonomy, i.e. the concepts, is formally defined in the ontology).
3. Implementation of the ontology (using a formal language to build the ontology in detail, which means even adding relationships between concepts, and rule axioms).
4. Evaluation of the ontology (testing the ontology to see if it meets the requirements).

The development process may not be linear, and several refinements can be made by repeating some of these stages (Roussey et al. 2011). Many different methodologies are available; however, the enterprise ontology development methodology defined by Grüninger and Fox (1995) is closely examined for the development of the required ontology.

Taking this into account, the methodology adopted in this research is presented below in Figure 13:

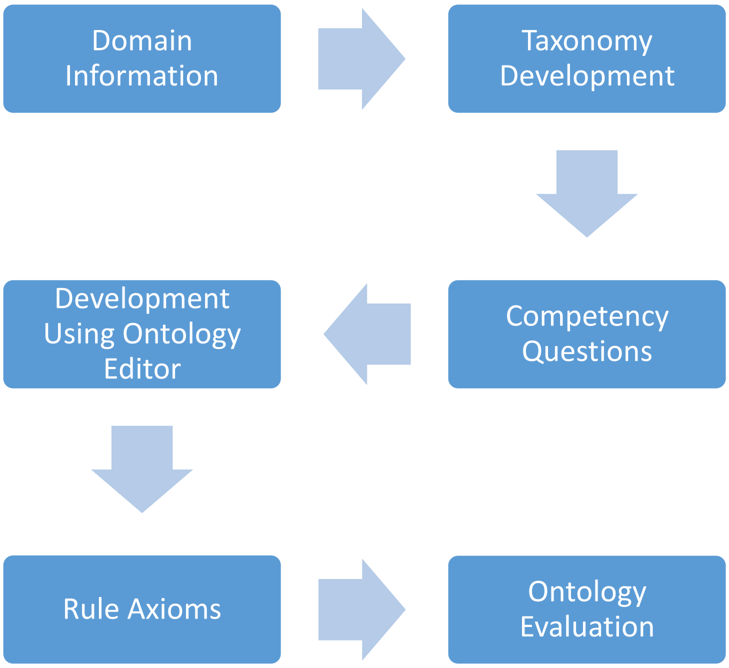



**Figure 13. Methodology adopted for development of REMO ontology**

### 3.1.1. Domain information collection

Here the domains are studied in detail to understand how the ontology can be applied for real-time energy management. This PhD adopted action research to understand the application of real-time energy management techniques in the domain. Action research further helped conceptualise the domain. Action research for this stage can be defined as: "*A disciplined process of inquiry conducted by and for those taking the action. The primary reason for engaging in action research is to assist the 'actor' in improving and/or refining his or her actions.*" (ASCD 2016). It is also meant to be a reflective process of progressive problem-solving (Culatta 2015).

The author was involved in two European Union Framework Programme 7 (FP7) research projects, which have been used as case studies in this process to understand the practical solutions applied to today's real-time energy management issues which are at the cusp of

European research. The two projects were SportE2[4] and Resilient[5]. SportE2 looked into demand side optimisation, whereas Resilient looked into supply side optimisation. Although the two projects were focused on different domains, demand and supply, both aimed to address the problem of real-time energy management using AI concepts. In the case of the demand side (within buildings), the primary focus was to reduce demand and costs while trying to maintain occupants' comfort levels. In the case of the district (supply side), the main objective was to keep carbon emissions and operational costs of the district at a minimum while trying to meet the demand. This phase also aimed to identify factors that affect the objectives of the district or building optimisation problem – costs, emissions, comfort, and efficiency. The artificial intelligence technologies used for these projects were very different in nature. Each project conducted a detailed literature review and selected some of these artificial intelligence techniques and algorithms.

### 3.1.2. Taxonomy development

A taxonomy[6] here represents a classification scheme that organises controlled vocabulary into a hierarchical structure based on the user needs. Through the action research and domain conceptualisation stage, key concepts are identified from the demand and supply domains that are needed for real-time energy management and optimisation.

Understanding the methodology in which artificial intelligence was used for real-time energy management was important so that taxonomy of **REMO** ontology could be appropriately designed to support these AI techniques. Moreover, the knowledge behind the use of these artificial intelligence models needed to be captured in **REMO** ontology through rule axioms. Capturing this knowledge meant that, once the ontology was built and instantiated, it would be capable of providing all the information needed to develop AI models. For example, in the case of prediction or optimisation models, the input parameters and output parameters can be identified.

The projects and the work undertaken by the author and colleagues as a part of this are presented in Chapter 4. The key project use cases were also beneficial to model the domain (supply and demand) related concepts in the ontology. The concepts identified here are listed in a hierarchical structure (class hierarchy) also establishing the relationships between various classes and concepts. Chapter 4 towards the end represents

---

[4] http://www.sporte2.eu/
[5] http://www.resilient-project.eu/

[6] http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html#sect-taxonomies

some of the learning and understanding from these projects which have helped develop the **REMO** taxonomy. Working on these projects as case studies helped with the following research questions, which were listed earlier in Chapter 1:

- What are the artificial intelligence techniques that can be applied at a building and district level for real-time energy management?
- What are the various objectives that need to be considered for energy management when it comes to building and district energy management?
- How can building and district energy management techniques be harmonised to take a multi-objective approach?

### 3.1.3. Competency questions

It was necessary to list competency questions as this helped to determine the scope of the ontology and some of its applications. The ontology-based knowledge-base should be able to answer these questions either through inferencing or querying (Grüninger and Fox 1995). Competency questions help define the scope of the ontology. They are usually just a sketch, and the list does not need to be exhaustive (Noy and McGuinness 2001). These are defined for **REMO** in Chapter 6 under Section 6.2.1.

### 3.1.4. Development using an ontology editor

The ontology was built using the TopBraid Composer Standard Edition[7], which is a powerful tool used to build semantic web and linked data applications. Web Ontology Language (OWL), which is an extension to RDFS (Resource Description Framework Schema), is a formal syntax for defining ontologies as simple classes and objects (Linked Data Tools 2015). Both RDFS[8] and OWL are W3C specifications. RDFS is a semantic extension of RDF through which a group of related resources (RDF) and the relationship between them can be described. RDF is a way of modelling data in triples – subject (denotes the resource), predicate (expresses the relationship between the subject and the object) and object. RDF and OWL[9] language use classes and subclasses to classify things regarding semantics. Classification of individuals into groups sharing common characteristics can be defined as a class in OWL Specification. Once all these classes are classified and the relationships between them have been established, a domain model is

---

[7] http://www.topquadrant.com/tools/ide-topbraid-composer-maestro-edition

[8] https://www.w3.org/TR/rdf-schema/

[9] https://www.w3.org/TR/owl-features/

complete where a common vocabulary and a shared understanding is established. Consequently, the instances of these classes and subclasses can be defined and are called individuals. Individual members of the class come under the semantic classification given by the OWL class. The individuals are related by properties:

(1) *Object properties* relate individuals or instances of two OWL classes and,
(2) *Datatype properties* relate individuals of OWL classes to literal values.

Defining all the individuals and their properties completes the instance model.

Reusing an existing ontology (which is well established in the domain) is recommended in ontology engineering to avoid "reinventing the wheel" (Madrazo and Sicilia 2014). These third-party ontologies can be imported to **REMO** ontology, and their concepts can be either reused or mapped with the similar **REMO** concepts.

### 3.1.5. Defining rule axioms

Ontologies are based on description logics, which is classical logic, whereas rules are based on logic programming. Rules, therefore, provide high expressivity with efficient reasoning support. Therefore, rules are usually combined with ontologies and can be used for problem-solving.

*Background on SPIN and SPARQL*

A wide range of business rules can be represented through SPIN rules. SPIN rules store SPARQL queries and help specify constraints or rules (World Wide Web Consortium 2011). SPARQL query stands for SPARQL query and RDF Query Language. It is a semantic query language, which is capable of retrieving and manipulating data stored in RDF format from databases. These queries are made possible using HTTP protocol, and they are sent from a client to a SPARQL endpoint. Because the interaction between the endpoint and the client happens through a machine-friendly protocol – it cannot be interpreted by humans – hence SPARQL requires an interface through which queries can be entered, and results can be displayed in a meaningful way (World Wide Web Consortium 2008).

SPIN rules also help define RDF class description properties –

- spin:rule – the inference rules can be defined here.
- spin:constraint – these can be used to define conditions that all members of a class need to meet.

- spin:constructor – any new instance that needs a default value can be defined through a rule using this property.

SWRL rules can also be used for the same purpose as SPIN rules. SWRL stands for Semantic Web Rule Language and is a combination of the OWL web ontology language and ruleML sublanguage of rule mark-up language. The rules are written in abstract syntax, and they make sure that a certain action is followed if certain other conditions happen (World Wide Web Consortium 2004).

SPIN rules were chosen as the rule engine for **REMO** ontology. One of the reasons for this is that the SPIN-related class properties are class-specific properties whereas SWRL is applied to the entire ontology. SPIN, being based on SPARQL, is more expressive that SWRL and hence is considered to be far superior than SWRL. SPIN rules are also capable of expressing constraints, defining new functions and templates. (World Wide Web Consortium n.d.). Another reason for preferring SPIN over SWRL was that numerous engines and databases could support SPARQL, and it is considered to be well established.

### 3.1.6. Ontology evaluation

There are many ways to validate ontologies, as shown by previous works (Staab and Studer 2009; Brank et al. 2005; Denny Vrandecic 2010). Ontology validation is important to prove that the ontology built is credible and is worth being reused (knowledge artefacts are reusable) and extended in the future. Evaluation is especially important today because an increasing number of ontologies are being built and users need to know which ones are credible enough to be reused.

The ontology evaluation can be defined to include two concepts as mentioned by Vrandecic 2010– verification and validation. Ontology verification is needed to see if the ontology has been built correctly, whereas ontology validation is needed to determine if the correct ontology has been constructed. For the **REMO** ontology a number of evaluation steps have been taken for both these types of validation, which have been adopted from the survey conducted by Hlomani and Stacey (2014):

<u>Verification</u>

Here, the correctness of the ontology is checked. The ontology is validated mainly through consistency checking, which looks for syntax-related issues and other violations regarding the range of parameters and axioms. It is therefore used for testing and debugging the ontology (both domain model and instance model). Doing this prevents ambiguous results being returned from the ontology when it is queried or reasoned.

Validation

Hlomani and Stacey (2014) also mentions 'application-based evaluation' as an important approach for evaluating ontologies, where the effectiveness of the ontology is assessed in the context of its application or use cases. To do this, an instance model of the ontology was also defined where ontology classes are instantiated with individuals. This instance model then undergoes a series of evaluation methods:

- The reasoning of the **REMO** ontology is tested by validating the relevance of inferred knowledge. Reasoning engines are algorithms that use the defined ontology and derive (or infer) new knowledge from it. These were not explicitly mentioned in the ontology during the instantiation process. Reasoning also helps identify any inconsistencies in the ontology.
- SPARQL queries are also used to test the instance model to see the relevance of the responses. The competency questions are evaluated here, and the relevance of the responses are analysed. Fox et al. (1997) in their work on **TOVE** ontology, stressed the importance of competence during validation where the ontology is explored in the context of their competency questions.

Various other metrics and validation approaches are also mentioned in this paper; however, in the case of **REMO,** the above-mentioned steps were chosen for validation. These steps test its basic working from an application point of view and quality regarding syntax and semantics.

This part of the research help draws conclusions relating to the final two research questions:

- Can ontologies help facilitate the harmonisation of demand and supply side optimisation? Moreover, how?
- Can the knowledge behind the optimisation models be captured in the ontology so that they can be replicated for similar districts and buildings?

## 3.2. Notation Conventions

Any referral to the ontology is highlighted by **bold Courier New** font. An entity or attribute of the ontology itself is denoted by Courier New font. The best practices adopted for naming and vocabulary are presented below (Structured Dynamics 2014):

- Name of classes starts with a capital letter. Classes in an ontology that have names consisting of more than one word are named using CamelCase Notation (no spaces left between the words). These classes represent the main concepts of the ontology. Most of these classes are also named as single nouns.
- The attributes (properties) in the ontology are named as verb senses – for example, *hasName*. These predicates are named starting with a lower-case letter, using mixedCase notation, and are italicised. Once again, no spaces are left between the words, and a capital letter is used for every word after the first.
- No particular naming conventions are applied for the individuals, i.e. the instances of the class. However, their names are italicised.
- In figures representing parts of the ontology, the grey shaded boxes refer to classes, and white boxes represent the individuals. The attributes of classes are represented in grey boxes with dashed line borders, whereas those of individuals are represented in white boxes with dashed line borders.

# 4. Action research through European Union research projects

This chapter covers in detail the action research conducted and knowledge concluded from working on the project case studies. This knowledge was fundamental to building **REMO** ontology as mentioned earlier in Section 3.1.1 In this Chapter, Section 4.1 looks into demand side energy management (building) – SportE2 – and Section 4.2 looks into production side energy management (district) – Resilient. Section 4.3 finally concludes some of the knowledge gained from action research and how the gaps identified in action research would be addressed thorough the **REMO** ontology.

## 4.1. SportE2 Project – demand side energy management

The aim of the SportE2 Project was to "*develop an integrated, modular, and scalable ICT system to manage energy consumption, generation, and exchange locally and within the larger context of the smart grid/neighbourhood.*" (SportE2 Project Consortium 2014). SportE2 aimed to use artificial intelligent techniques such as multi-objective optimisation and neural network model (ANN) for real-time energy management in sports facilities. The aim of the project was to cut down the energy consumption of existing sports facilities by 30%. The problem as a whole in the facility was broken down into use cases and scenarios following the divide and conquer principle. The rationale behind this was that small savings in each of these different scenarios would bring about a significant overall energy saving in the facility.

The project developed four scalable modules which could either be integrated or adopted separately based on the client's requirements. The modules were (shown in figure 14 below):

- SportE2 HOW (smart metering to determine where energy is being consumed),
- SportE2 WHEN (integrated control systems that enable the actuation of energy sourcing and consumption),
- SportE2 WHY (intelligent and optimal decision-making given smart metering data and control capabilities), and
- SportE2 WHERE (a multi-facility management portal).

The interaction between the different modules is shown below in Figure 14 (SportE2 Project Consortium 2014).

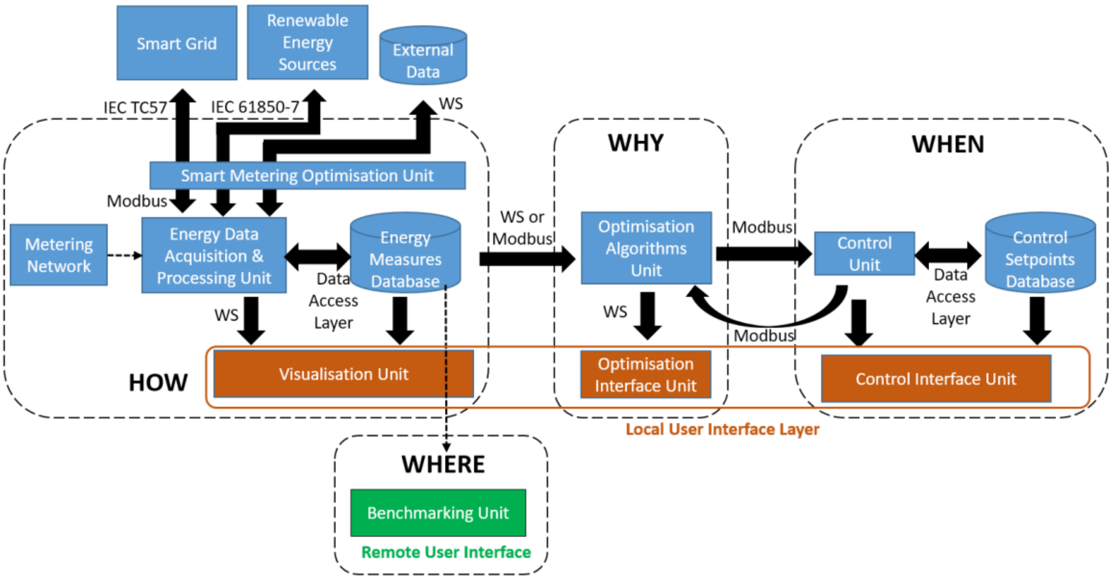



**Figure 14. SportE2 System Architecture (SportE2 Project Consortium 2014)**

*WHY module workflow*

Cardiff University was in charge of the development of the WHY module, and this module relies on real-time data, which is collected through sensors and meters and fine-tuned by the HOW module. The data is made available via an automation server, provided by the WHEN module. The WHY module requests data every 15 minutes and the optimisation system developed by Cardiff University consequently aims to provide optimised control parameters. The optimisation objectives in most cases were to minimise energy and maximise comfort. The WHY module uses a range of artificial intelligence models (artificial neural network models, and multi-objective optimisation algorithms) and mathematical models (EnergyPlus or Simulink simulation models). The entire building optimisation problem was split into use cases and scenarios, each of which had its prediction and optimisation model. The initial stages of the project finalised these use cases, which also had to be compatible with the other modules – HOW and WHEN.

The optimisation solution adopted by the WHY module is shown below in Figure 15. Simulation models are initially used to run energy simulations for various scenarios and use cases. These results are then used to train artificial neural network (ANN) models for each use case or scenario. Once the ANN model is trained and validated, different optimisation algorithms can be used to run with it as the cost function.

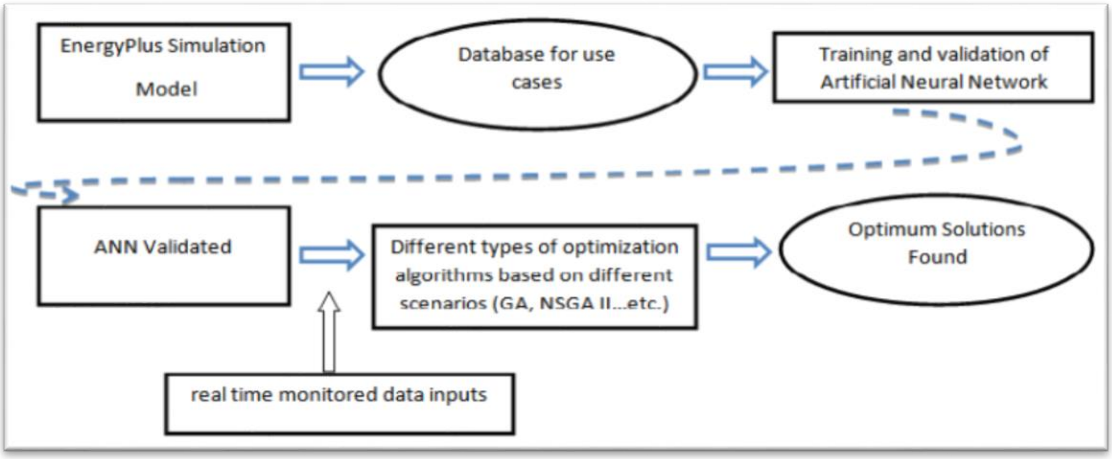



**Figure 15. ANN embedded within optimisation Framework**

On receiving the data, the WHY module aimed to return optimum solutions to the automation server within a couple of minutes, and hence High-Performance Computing (HPC) was implemented. The details of the HPC and its integration with the web services and optimisation are mentioned in Petri et al. (2014a).

#### 4.1.1. Contribution to SportE2 - Use case and scenario development

The authors major contribution in the project was the development and finalisation of these use cases and scenarios. Specific operational use cases and scenarios (instead of a comprehensive map of all relevant variables) were developed mainly to limit the amount of variables being considered in an optimisation problem. Further analysis of the scenarios was also carried out to limit variables by focusing on the most ‘sensible’ (to the multi-objective optimisation problem) ones. A general template was developed by the author to represent the WHY module use cases and their requirements is presented later in the section. The template was developed by interacting with pilot owners and facility managers. The methodology for the development of use cases is also presented later in this section. SportE2 WHY optimisation can be applied to various possible use cases and scenarios in a facility, but its feasibility depends on:

1. Cost constraints – Cost implications of the practical implementation of the solutions should be discussed with pilot owners before making decisions on use case implementation. A list of sensors, actuators, meters, and other equipment needed and their costs should be finalised for each scenario before pilot owners make decisions.

2. Energy Audits – Energy audits of the facility help identify the zones in the facility, which are major energy consumers. Use cases/scenarios, therefore, could focus on these areas.
3. Practicality – Implementation of SportE2. Solutions would need the scenarios to be feasible for all SportE2 modules especially HOW (monitoring) and WHEN (control) modules. Sufficient data from sensors and control of devices and actuators must be feasible in reality.

As mentioned earlier in this section, simulation models are key to SportE2 WHY module to be able to produce enough dataset for ANN training. However, there can be cases when simulation models of buildings do not exist, or the buildings could be too complex to be modelled. In such situations the BMS can be used to provide historical data and this data can be used to train the ANN model consequently. Figure 16 shows the generalised methodology utilised in SportE2 WHY to finalise scenario definitions for different pilots.

**1** Generic list of use cases

- General list of use cases and scenarios defined.
- Generic definition of use cases.

**2** Pilot specific scenarios

- Each pilot studied separately.
- Scenarios identified for specific pilots.

**3** Simulation models or historical data

- Simulation models of each pilot studied in detail.
- Historical data investigated if simulation model is not available.

**4** Mapping of devices and variables

- Sensors, meters and actuators from scenario definition mapped to simulation model or historical data.
- A monitoring plan is proposed for data to be collected from any new installations.

**5** Final decision on scenario feasibility

- Decision made on feasibility of scenario for SportE2 applications.
- Agreement reached between the different partners.

**6** Fine-tuning of scenario

- List of sensors, meters and actuators finalised for installation.
- Scenario fine-tuned accordingly.

**7** Extraction of data

- Data extracted from results of simulation.
- Simulation models are run iteratively to collect data from different possibilities.

**8** Results analysis

- Unwanted data removed from results.
- This ensures safe learning of the neural network.

**9** Final data set prepared and ANN training

- Final data set prepared and classified as inputs and outputs for training purposes.
- ANN model trained accordingly.

**Figure 16. Use case development methodology and actions**

Figure 16 is explained below:

1. Initially, a general list of use cases and scenarios was identified based on literature and domain knowledge. It included a full list of potential scenarios for real-time energy management that could be applied in sports facilities during the operational stages.
2. The pilot-specific scenarios were identified by interaction with pilot owners and facility managers. The energy audit was also taken into consideration at this stage. The HOW and WHEN module technical partners were consulted to decide on a final list of scenarios to be implemented.
3. Simulation models and historical data were analysed to check if they could provide sufficient data for training of ANN models for the respective scenarios.
4. The devices, sensors, meters and actuators in the facility were mapped with variables defined in the WHY module as input and output for each use case/scenario. Knowing this helped analyse the feasibility to implement the scenario in reality. The scenario would be then fine-tuned based on available devices and variables or, in certain cases, the missing physical devices would be installed based on the requirements.
5. Various simulation runs were performed by considering the broad range of possible solutions (using a range for decision variable) to gather data, in the cases when historical data is not available.
6. The data was analysed, and unwanted data, which was not relevant to the use case/scenario, was removed. For example, data related to unoccupied hours can be removed if not related to the scenario. The processed data sets were consequently used for ANN training

The different use cases developed in the project by Cardiff University, which uses ANN-based optimisation, is described in the project deliverable (Cardiff University 2013). However couple of sample use cases are shown below.

*Use case examples for FIDIA and EMTE pilot*

Evaporation in swimming pools is one of the root causes of them being major energy consumers. In the FIDIA pilot, the audits suggested that the swimming pool was responsible for almost 50% of electricity consumption, and 44% of thermal energy onsite. The scenario developed through the project proposes to optimise the air-handling unit in the swimming pool zone to ensure proper air treatment for the zone, and aims to provide sufficient conditioned air to the area to reduce evaporation of surface water. Supplied inlet air temperature was controlled to maintain comfort requirements while reducing energy usage. The scenario also has the potential to control the supply airflow rate if need be, but, in the case of the FIDIA pilot, control limitations meant this was not possible in reality.

**Table 2. Parameters Involved in Scenario**

| Use case 1: Optimisation of HVAC System in swimming pool room | | | | | | |
|---|---|---|---|---|---|---|
| **Scenario** | **Area** | **Objective** | **Variables** | | **Sensors/Meters/Set points** | **Units** |
| **Air treatment** | Swimming pool area | Minimise energy consumption; Maximise thermal comfort | ***Input for ANN & Optimisation*** | Occupancy | Occupancy sensor | - |
| | | | | Indoor temperature | Temperature sensor | deg. C |
| | | | | Water temperature | Temperature sensor | deg. C |
| | | | | Indoor humidity | Humidity sensor | % |
| | | | | Air temperature inlet | Temperature sensor | deg. C |
| | | | | Supplied air flow rate | Velocity sensor | kg/s |
| | | | ***Output of ANN*** | PMV | - | - |
| | | | | Electrical energy | Electricity meter | kwh |
| | | | | Thermal energy supplied | Heat meter | kwh |
| | | | ***Output of Optimisation*** | Optimised air temperature inlet | Optimised setpoint | deg. C |
| | | | | PMV | - | - |
| | | | | Optimised electrical energy | - | kwh |
| | | | | Optimised thermal energy supplied | - | kwh |
| | | | | | | |
| | | | ***additional parameters for Validity Check*** | Carbon concentrations | Co2 sensor (air quality) | ppm |
| | | | | Chlorine in air | Cl sensor (air quality) | ppm |
| **Actors** | Automation server, facility technician, sensor, actuator. | | | | | |
| **When Applicable** | During operational period | | | | | |
| **Additional Notes** | 1. By default all input parameters are validated before any optimisation can take place. In FIDIA scenario 1, there are two extra parameters which needs to be considered for validation. Variables like CO2 concentrations and Chlorine concentration in air can affect the air quality of room. Therefore, WHY module takes into account these variables even though it is not used in the ANN model nor for optimisation.<br>2. Here, although the air flow rate used is kg/s in WHY module, the sensors measure air velocity in m/s in FIDIA facility. This air velocity is used to calculate the air flow rate by multiplying with density of air and duct size. | | | | | |

Table 2 above shows the parameters involved in the optimisation process. Implementation of the optimisation scenario, in reality, is shown below in Figure 17.

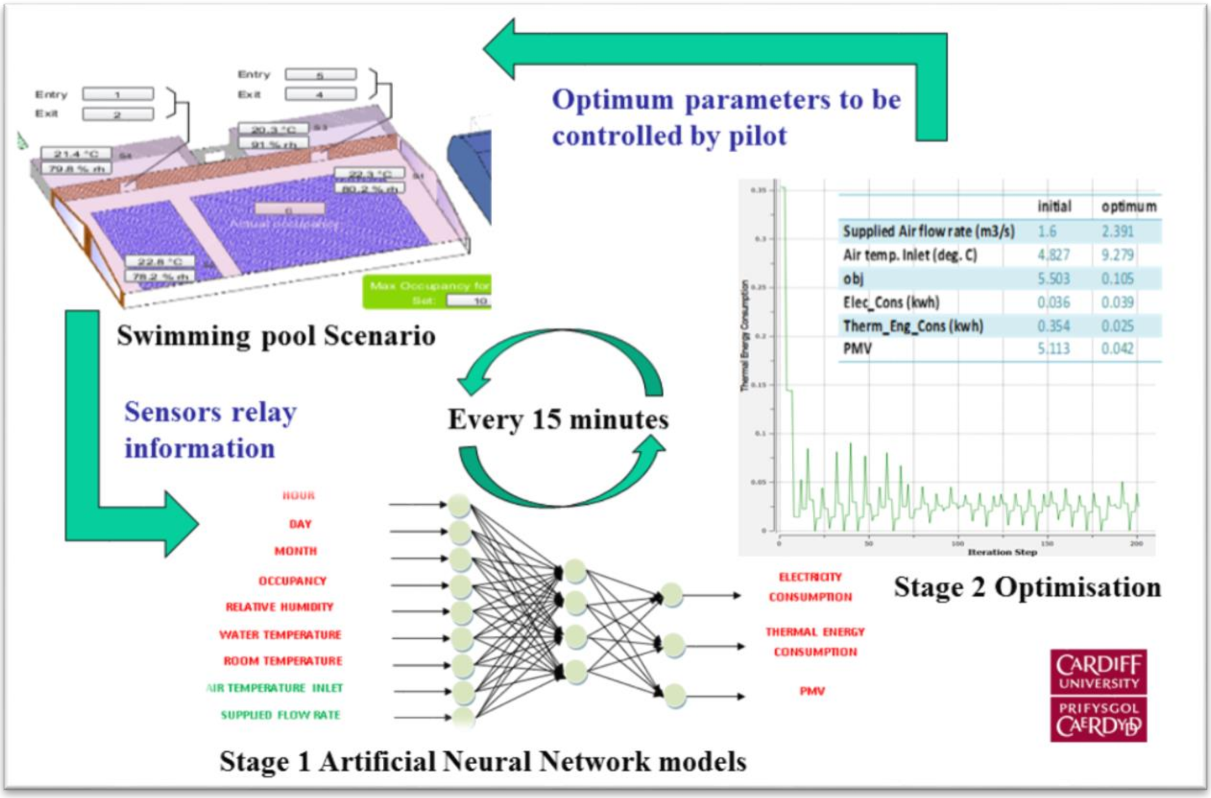



**Figure 17. Sample use case of SportE2**

Figure 17 shows the workflow of WHY module where initially, the BMS provides real-time readings from sensors and actuators to be passed on as the inputs of the ANN model for each scenario. The optimisation model then finds the optimum setpoints for the scenario by running the ANN model repeatedly (Yang et al. 2014). Moreover, some variables like $CO_2$ concentrations and chlorine concentration in air can affect the air quality of a room; these are variables that are not present in the optimisation model or ANN model, but simple rules are implemented in the optimisation for this scenario wherein drastic measures are taken to counteract the situation. For example, in the case of high chlorine in the air, the supply airflow rate is increased to maximum, and no optimisation is performed until the following time step.

Similarly, an optimisation scenario was also developed for HVAC systems in fitness rooms. This use case was developed for the EMTE pilot. Table 3 below shows the variables needed for prediction and optimisation models. Other use cases and scenarios developed can be found in the deliverable <reference sporte2 deliverable>

**Table 3. EMTE pilot use case 1 for optimisation of HVAC system**

| EMTE Use case 1: Optimisation of HVAC System | | | | | | |
|---|---|---|---|---|---|---|
| **Scenario** | **Area** | **Objective** | **Variables** | | **Sensors/Meters /Setpoints** | **Units** |
| **Optimisation of HVAC system (air handling unit)** | Fitness Room 1 | Minimise Energy consumption; Maximise Thermal comfort | ***Input for ANN & Optimisation*** | Occupancy | Occupancy sensor | - |
| | | | | Outdoor temperature | Temperature sensor | deg. C |
| | | | | Indoor temperature | Temperature sensor | deg. C |
| | | | | Activity type* | (no sensor) based on schedule | - |
| | | | | Indoor humidity | Humidity sensor | % |
| | | | | Outdoor humidity | Humidity sensor | % |
| | | | | Co2 concentrations | Co2 sensor | ppm |
| | | | | Thermostat heating setpoint | Setpoint | deg. C |
| | | | | Supplied air flow rate | Velocity sensor | kg/s |
| | | | ***Output of ANN*** | Pmv | - | - |
| | | | | Fancoil heating energy | Heat meter | kwh |
| | | | ***Output of Optimisation*** | Optimised supplied air flow rate | Optimised setpoint | kg/s |
| | | | | Optimised thermostat heating setpoint | Optimised setpoint (through thermostat) | deg. C |
| | | | | Pmv | - | - |
| | | | | Optimised fancoil heating energy | - | kwh |
| **Actors** | Automation server, facility technician, sensor, actuator. | | | | | |
| **When Applicable** | During operational period (winter months) | | | | | |
| **Additional Notes** | 1. *Activity type is an input to ANN model and this is based on a pre-defined schedule by pilot facility managers. The activity types are: *Aerobics – activity type 5; Pilates – activity type 4; GAP (exercises to strengthen legs and abdomen) – activity type 3; GIM (elderly people) - activity type 2;* and *Aerotxiki (Aerobics for children) – activity type 1.*<br>2. PMV computation is from EnergyPlus model. In reality this cannot be calculated. PMV gives a rough idea of comfort of occupants. This is not monitored by any sensor or meter.<br>3. Here, although the air flow rate used is kg/s in WHY module, the sensors measure air velocity in m/s in FIDIA facility. This air velocity is used to calculate the air flow rate by multiplying with density of air and duct size. | | | | | |

#### 4.1.2. Results achieved through SportE2 optimisation

The ANN model development and its testing are presented by Yuce et al. (2014). The details of the optimisation framework of SportE2 and its testing can be found in work presented by Yang et al. (2014). Use of high-performance computing and cloud-based services for optimisation is detailed by Petri et. al (2014a). The SportE2 solutions were very efficient and have proven energy savings in various real-world scenarios, as shown below through the project validation. The objective of the project was to guarantee energy savings of about 30%.

The energy validation phase of the project was led by project partners, and they analysed data in two stages for each of the pilot. Initially the SportE2 systems and solutions were turned off for a period (typically a working week). During this time, plants and devices are controlled manually, and the energy consumption was monitored in detail using the various sub-meters available to the pilot. Consequently, in the second stage, the SportE2 systems were activated (i.e. optimisation module is activated) and data was monitored. A comparison was made between the two stages and the average energy savings for the different scenarios achieved, when the system was active, is given in detail (Università Politecnica Delle Marche 2014) below. The energy savings were isolated based on the different optimisation scenarios.

**Pilot 1 – FIDIA, Italy**: Here the swimming pool AHU optimisation scenario was implemented (shown in table 2) and testing was carried out for the time period as shown in table 4 below.

**Table 4. Testing period for FIDIA pilot.**

| | Winter | Spring | Summer |
|---|---|---|---|
| **SportE2 OFF** | 08/02/2014 - 14/02/2014 | 09/04/2014 - 15/04/2015 | 18/06/2014 - 24/06/2014 |
| **SportE2 ON Week 1** | 22/03/2014 - 28/03/2014 | 07/05/2014 - 15/05/2014 | 02/07/2014 - 08/07/2014 |
| **SportE2 ON Week 2** | 29/03/2014 - 04/04/2014 | 21/05/2014 - 27/05/2014 | 23/07/2014 - 29/07/2014 |

The results of monitoring are shown below in table 5 and the average electrical and thermal savings achieved in the pilot through the swimming pool optimisation scenario was 34 % and 29 % respectively during the testing period.

**Table 5. Results achieved by SportE2 solutions in FIDIA pilot for the swimming pool scenario.**

| **Electricity Savings** | | | | | |
|---|---|---|---|---|---|
| **Testing Phase** | **Unit** | **Winter** | **Spring** | **Summer** | **Average** |
| SportE2 OFF Week Energy Consumption (Baseline) | kWh/Week | 1.983 | 1.92 | 3.3 | |
| SportE2 ON Week 1 Energy Consumption | kWh/Week | 1.365 | 1.275 | 2.2 | |
| *Savings (baseline Vs week 1)* | *%* | *31%* | *34%* | *33%* | *33%* |
| SportE2 ON Week 2 Energy Consumption | kWh/Week | 1.278 | 1.398 | 1.9 | |
| *Savings (baseline Vs week 2)* | *%* | 36% | 27% | 42% | 35% |
| **Thermal Energy Savings** | | | | | |
| **Testing Phase** | **Unit** | **Winter** | **Spring** | **Summer** | **Average** |
| SportE2 OFF Week Energy Consumption (Baseline) | kWh/Week | 11.5 | 8.4 | 3.5 | |
| SportE2 ON Week 1 Energy Consumption | kWh/Week | 10.6 | 6.7 | 2.3 | |
| *Savings (baseline Vs week 1)* | *%* | *8%* | *20%* | *34%* | *21%* |
| SportE2 ON Week 2 Energy Consumption | kWh/Week | 8.5 | 4.6 | 2.1 | |
| *Savings (baseline Vs week 2)* | *%* | 26% | 45% | 40% | 37% |

**Pilot 2- EMTE Sport, Spain**: This pilot implemented the air-handling-unit optimisation scenario for one of the fitness rooms in the facility. The average energy savings achieved here during the testing period here was 47% and 35% of electrical and thermal energy respectively as calculated from the results shown in table 6 below. Testing took place mainly during winter. During the spring season, only electricity related data could be monitored whereas thermal energy data was missing.

**Table 6. Results achieved by SportE2 solutions in EMTE pilot for the fitness room HVAC optimisation scenario.**

| Electricity Savings | | | | |
|---|---|---|---|---|
| **Testing Phase** | **Unit** | **Winter** | **Spring** | **Average** |
| SportE2 Off Week Energy Consumption (Baseline) | kWh/Week | 18 | 24.4 | |
| SportE2 On Week 1 Energy Consumption | kWh /Week | 7.3 | 17.4 | |
| *Savings (baseline Vs week 1)* | *%* | *59%* | *29%* | *44%* |
| SportE2 On Week 2 Energy Consumption | kWh /Week | 7.1 | 15.4 | |
| *Savings (baseline Vs week 2)* | *%* | 61% | 37% | 49% |
| **Thermal Energy Savings** | | | | |
| **Testing Phase** | **Unit** | **Winter** | **Spring** | **Average** |
| SportE2 Off Week Energy Consumption (Baseline) | kWh /Week | 179 | n/a | |
| SportE2 On Week 1 Energy Consumption | kWh /Week | 106 | n/a | |
| *Savings (baseline Vs week 1)* | *%* | *41%* | | *41%* |
| SportE2 On Week 2 Energy Consumption | kWh /Week | 129 | n/a | |
| *Savings (baseline Vs week 2)* | *%* | 28% | | 28% |

Note: the testing period here was slightly different to that of the FIDIA pilot. Further testing results and analysis are provided in the project validation report which is available from Sporte2 website online[10]. On average SportE2 solutions helped achieve 36 % energy savings across the three pilots.

### 4.1.3. Knowledge processing from SportE2 contributing to **REMO** ontology

Some of the highlights and knowledge gained from this project were:

- SportE2's generic and scalable smart energy management system had embedded intelligence (i.e. contextual understanding of the scenario-based dependent and independent governing variables and their complex interactions) which addressed the limitations of existing SCADA-based commercial energy systems. SCADA-based systems only provide the basic logic-based control capability; it depends on a pre-defined static schedule or setpoints for control of key equipment with the hope of achieving the required comfort level and energy-saving target. The

[10] http://www.sporte2.eu/public-documents/

achieved results are usually not satisfactory because the static schedule/set points cannot factor in the continuously changing environmental and building usage condition. SportE2, on the other hand, uses the real-time data to react to those changing conditions. The project reinforces the need for real-time energy management. Moreover, the discussions held with facility managers and pilot owners over the course of working on this project reaffirmed the need for real-time energy management in buildings, and in this particular case for sports facilities, because of the increasing number of potential use cases for real-time energy management.

- Energy optimisation for the entire building could be complex, and hence a divide and rule approach was adopted here, where use cases and scenarios were developed to be applied for individual zones of the building. It was clear that the **REMO** ontology developed should be able to support these use cases and scenarios, and hence the concepts relevant for this had to be modelled in **REMO**.
- The results of the project imply that the solution adopted in the SportE2 WHY module specifically is a suitable methodology for real-time energy management. Some of the use cases developed in this project could be applied to other buildings as well which have similar optimisation problems.
- The authors used the Protégé[11] tool initially to model the concepts that were relevant for building energy optimisation through an ontology called **SportE2** ontology (Jayan et al. 2014). The idea behind this was to use this ontology to aid the optimisation process. However, this work had some limitations:
  1. It was focused at a building level, and no district level concepts were modelled for a holistic optimisation.
  2. Although use cases were modelled into the ontology, the work was incomplete as it could not show the working of these use cases.

However, a few of the **SportE2** ontology concepts were adopted into the **REMO** ontology, as explained later in Section 4.3.

## 4.2. Resilient project – supply side energy management and optimisation

This section looks into the author's work on the Resilient project and also on using numerical optimisation techniques for supply side optimisation, similar to the work

[11] http://protege.stanford.edu/

developed in SportE2, but at a district scale. This section is sub-divided into two: Section 4.2.1 studies the ontology developed to support the Resilient framework and Section 4.2.2 looks into details of the numerical optimisation applied for supply side energy management.

The aim of the Resilient project was to "*design, develop, install and assess the benefits of a new integrated concept of interconnectivity between buildings, distributed energy resources and grids at a district level*" (Resilient Project Consortium 2012). The project therefore required new ICT components adapted to the context of energy management at the district scale. Cardiff University's role in the project was to develop a district information model (Hippolyte et al. 2014) using semantic models. The district information model was developed using ontologies that would enable it to adapt to a broad spectrum of technologies including new energy supply and building technologies as suggested by Keirstead et al. (2012). The district information model is then further used by multi-agent systems and simulation models to help increase energy efficiency in the district in real time, as shown below in Figure 18. The `ee-district` ontology and the associated tools developed are applied at a district level, and the primary aim is to optimise the local energy production and consumption solely in the district.

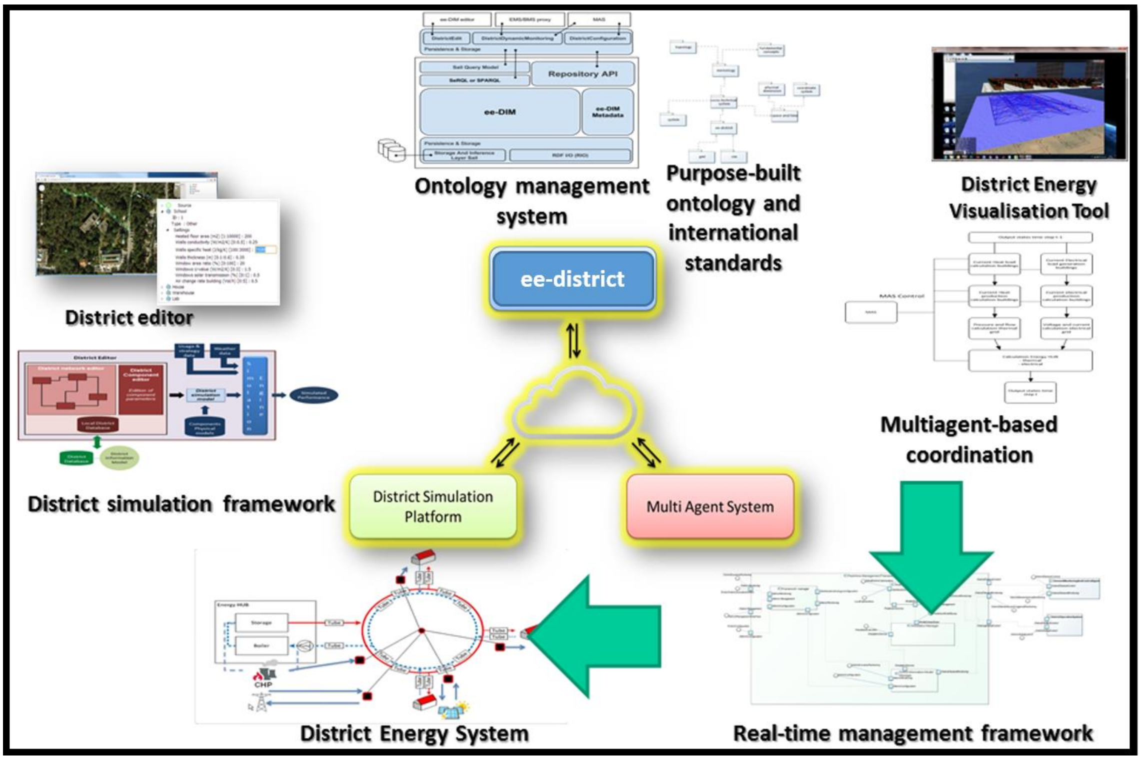



**Figure 18. RESILIENT project proof of concept**

### 4.2.1. `ee-district` ontology development

The district information model contained many domain-specific ontologies. The development of these domain-specific ontologies can be an ad hoc process (Madrazo et al. 2012). Although the strategy adopted for developing these ontologies, i.e. methodology, is case specific, the ontology itself did not need to be built from scratch. The ontology could have its structure and content constructed from a mix of both existing ontologies and other specific standards, which is the case in any modern software development project (d'Aquin et al. 2008). The core of the district information model is the district energy ontology named '**`ee-district`**'.

The **`ee-district`** meta-model *"formalize a generic, yet capable of specialisation, description of district elements as a socio-technical system. This formalisation allows then to produce machine-readable (and even machine understandable) models usable by software tools"* (Hippolyte et al. 2014, p.107). The multi-agent systems-based software tools in the Resilient framework also require the ontology to infer to and extract rules from it. A network of ontologies together forms the **`ee-district`** meta model which conceptualises the elements of the district energy system, their characteristics and relationships, and the various constraints of these systems. Using best practices of the semantic knowledge field, some of these ontologies were built from UML models. They were integrated into the network following novel alignment and modularisation methods.

*Methodology*

The author was mainly involved in the ontology development work conducted in the Resilient project, especially the development of the taxonomy of the domain ontology. The ontology was created in OWL language and was iteratively developed as knowledge collected throughout the different stages of the methodology. The development involved various stages of collaborative work and discussions between the different partners in the project. Some of the inputs were from (Cardiff University 2014):

- Literature from Cricchio et al. (2012).
- Questionnaires and interviews with stakeholders (reference the deliverable…).
- Domain knowledge gained from experts.
- Various standards such as IEC 61970-301:2011 (McMorran 2007; International Electrotechnical Commission et al. 2013).
- Existing ontologies.

The methodology for ontology development is described in Figure 19 below (Hippolyte et al. 2014):

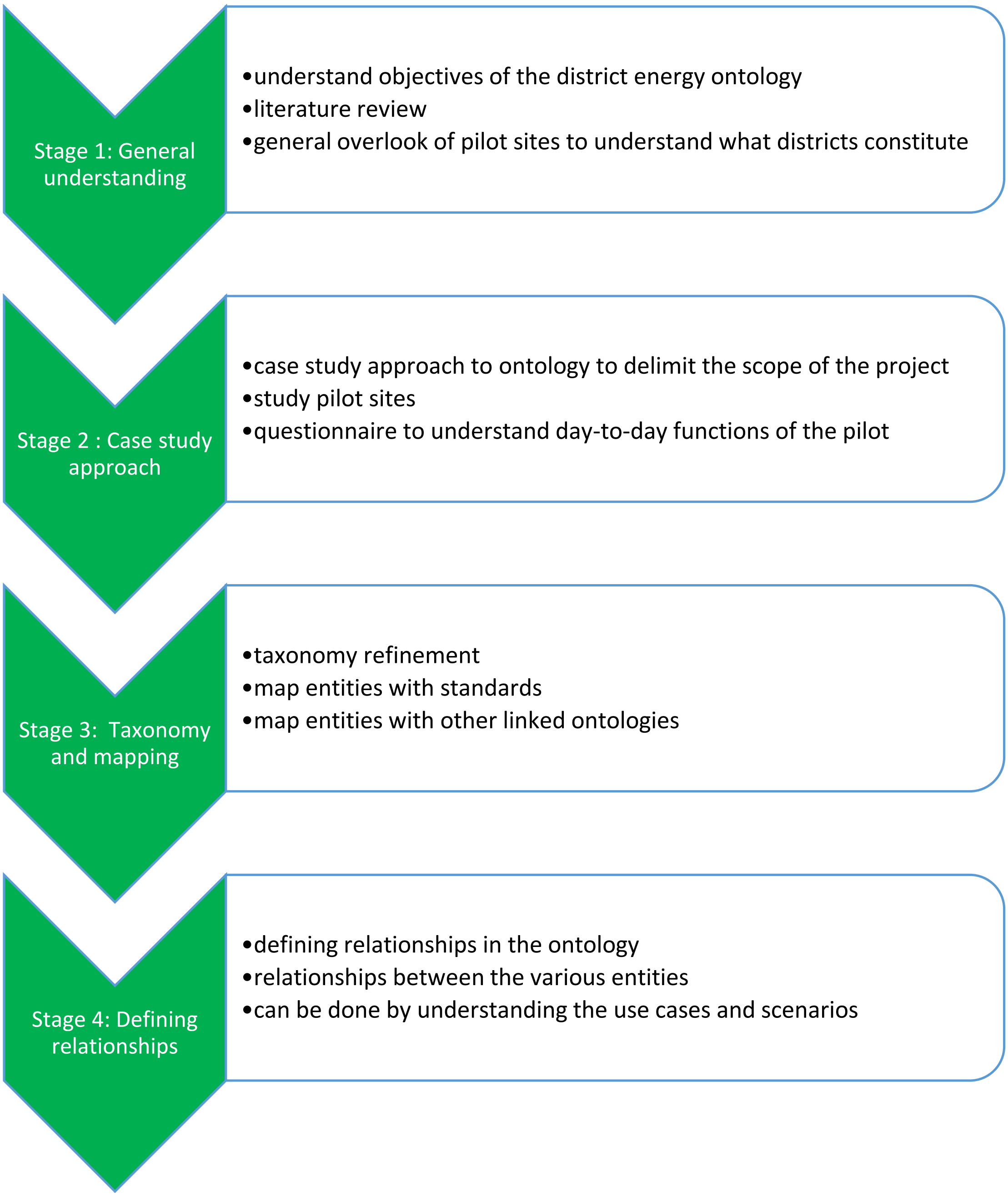



**Figure 19. The methodology adopted for ontology development**

Here the author was leading stages one, two and four. Stage three was mainly led by the authors colleague Dr. Jean-Laurent Hippolyte as it involved mapping the domain ontology with other standards and ontologies which were relevant to the domain.

**Stage 1:** the first step was to understand the use of the ontology in the scope of the project. The Resilient ontology was built for aiding MAS to coordinate the district energy systems, which involves a combination of storage systems, generation units, cogeneration units and energy users. Therefore, the ontology was required to:

- Answer queries from real-time optimisation software which incorporates the MAS;
- Ensure interoperability between the district coordination-level entities and building-level entities and/or energy system-level entities.

All the district entities – consumers, distributors and energy producers – are therefore modelled in the ontology. It also includes the district energy infrastructure (pipelines, power cables); the demand/supply load schedules; the overall system constraints; the objectives that are to be met; and the individual entities such as buildings, energy sources; etc.

This stage also involves gaining knowledge from the Resilient project's pilot sites. The project has three pilot sites – in Italy, Belgium and Wales. The different physical components installed in a district and the various stakeholders involved on the site are studied. Some of the key features of the district energy ontology concluded during this stage were:

• The ontology models energy information at a district level.

• It needed to support real-time decision-making for district energy optimisation.

• OWL Semantic Web Language, which provides the ontology engineers with a good extension of modelling formalisms, was chosen as the language for development.

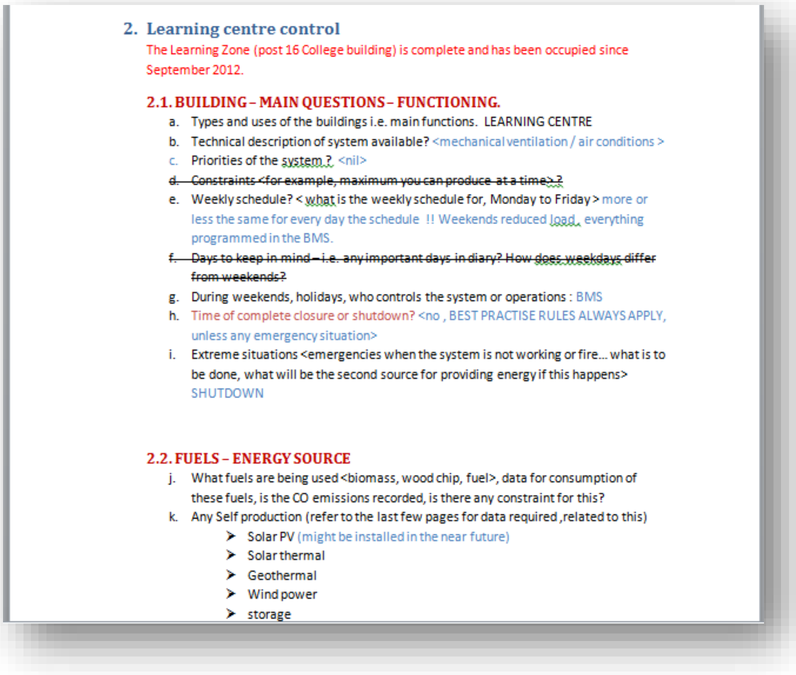
• The ontology should also be able to link with other standards and well-accepted ontologies in the domain.

Therefore, stage one of ontology development was largely based on literature review and general understanding of how a district works.

**Stage 2:** the second stage involved studying the pilot sites' day-to-day operations in the district. This stage is important because the working of the district in reality can be different to that learnt from literature or during the actual design stages.

A questionnaire was used as one of the key tools to source information during this stage. The questionnaire was aimed at the Welsh site first. The various sections of the questionnaire are directed at different consumers and producers in the pilot site. Figure

20 shows a screenshot of the questionnaire. The questionnaire prepared was answered through a series of interviews (both in person and over the telephone) and site visits. The Semanco project (Semanco Consortium 2011) also followed this approach of using questionnaires and interacting with stakeholders, as it helps limit the scope of the research and also helps identify potential use case and scenarios. These use cases and scenarios assist the development of the ontology from the application perspective (end use).



2. Learning centre control

The Learning Zone (post 16 College building) is complete and has been occupied since September 2012.

2.1. BUILDING – MAIN QUESTIONS – FUNCTIONING.

a. Types and uses of the buildings i.e. main functions. LEARNING CENTRE
b. Technical description of system available? <mechanical ventilation / air conditions >
c. Priorities of the system ? <nil>
d. ~~Constraints <for example, maximum you can produce at a time>?~~
e. Weekly schedule? < what is the weekly schedule for, Monday to Friday > more or less the same for every day the schedule !! Weekends reduced load , everything programmed in the BMS.
f. ~~Days to keep in mind – i.e. any important days in diary? How does weekdays differ from weekends?~~
g. During weekends, holidays, who controls the system or operations : BMS
h. Time of complete closure or shutdown? <no , BEST PRACTISE RULES ALWAYS APPLY, unless any emergency situation>
i. Extreme situations <emergencies when the system is not working or fire… what is to be done, what will be the second source for providing energy if this happens> SHUTDOWN

2.2. FUELS – ENERGY SOURCE

j. What fuels are being used <biomass, wood chip, fuel>, data for consumption of these fuels, is the CO emissions recorded, is there any constraint for this?
k. Any Self production (refer to the last few pages for data required ,related to this)
   - Solar PV (might be installed in the near future)
   - Solar thermal
   - Geothermal
   - Wind power
   - storage

**Figure 20: Case study questionnaire, sample questions**

In the future, a more efficient way of collecting and sharing information needs to be put forward, such as using any survey management systems (Bristol online survey[12]) or web-based wikis such as Cicero (Suárez-Figueroa et al. 2008)

**Stage 3:** the next stage dealt with defining the details of the entities in the ontology, which also involved grouping the entities and looking for associations between them (Hippolyte et al. 2014). During this stage, another action was to align the domain ontology with other existing standards or ontologies that were relevant to it. The alignment involves mapping the relevant concepts from `ee-district` ontology with the entities from other ontologies. This task can be tedious but increases the robustness of the ontology and also enhances the potential of the ontology to be reused.

[12] https://www.onlinesurveys.ac.uk/

**Stage 4:** based on how the ontology was going to be used, the relationships between its entities were defined during this stage. SWRL rules could also be implemented which will help the querying or reasoning process. As seen in the literature review, adding SWRL rules can also help in rule-based decision-making for a particular application such as energy management. Such practices are usually not machine-readable and therefore the district's individual energy management systems are not capable of responding to such scenarios by themselves.

*Conceptual model of ee-district*

The **`ee-district`** ontology contains concepts that are related to the district energy system. The district energy system contains physical entities and social entities. Physical entities would be buildings, energy networks, energy systems, storage facilities, etc., whereas the social entities would be mainly the stakeholders and the contracts they have with the physical entities. It was important to cover both these entities and their relationships in the ee-district ontology for the purpose of optimisation. Figure 21 below shows a very simple example of how the physical and social concepts are aligned in the **`ee-district`** ontology. The horizontal links in the figure depict the relationships between the entities in the same domain, i.e. between entities that are network specific. The vertical dotted lines show the cross-domain relationships. A socio-technical ontology developed by Koen van Dam ( 2009) was reused (as an imported ontology) in **`ee-district`** to describe the relations between social and technical concepts. The socio-technical ontology is described in detail in the next section, which explains the semantic structure and other standard ontologies that were reused in **`ee-district`**.

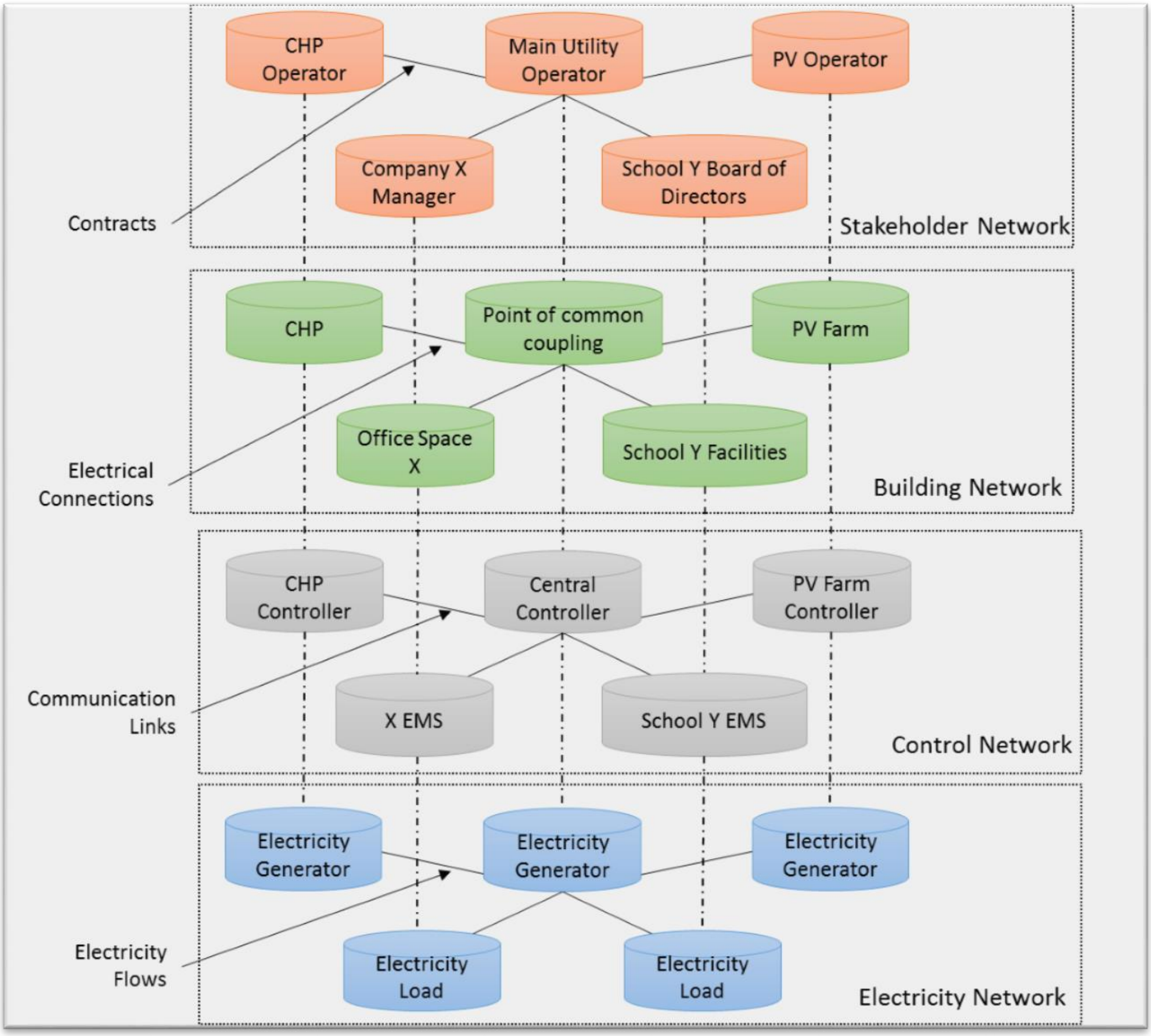



**Figure 21. Sample concepts and relationships in the `ee-district` ontology**

*Semantic structure*

Building and energy sectors are fragmented and interdisciplinary in nature and therefore any knowledge management system looking to aid practitioners in these areas need to adopt a layered and modular approach (Rezgui 2007). The **`ee-district`** ontology structure, therefore, had two main concepts:

- The district energy systems being modelled as socio-technical systems.
- Bringing together a set of diverse ontologies to form the meta-model of the district energy system.

Dr. Jean-Laurent Hippolyte worked on mapping the domain ontology with these various other ontologies. The various ontologies contained in this meta-model are shown below:

1. Socio-technical ontology

Koen van Dam (2009) abstracted the concepts of social and technical systems across disciplines. The work introduces the concept of nodes that represent the systems of the network that contain both social and technical elements. The social nodes are involved in the decision-making of physical nodes, and the physical nodes represent the actual elements of the physical world.

Figure 21 above shows an example where the district energy systems are an instantiation of a socio-technical system. Here, the top layer represents the social elements (such as owners and stakeholders) and the physical elements are represented in the three layers below (such as buildings and energy sources). The socio-technical systems also have a hierarchy of classes – physical edges and social edges (see Figure 22 below). The physical edges connect the physical nodes whereas the social edges connect the social nodes.

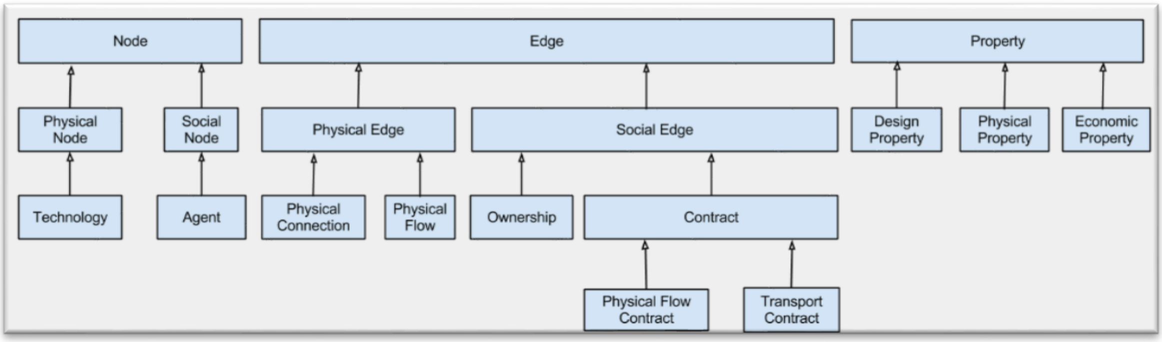



**Figure 22. Hierarchy of socio-technical systems ontology (adopted from van Dam 2009)**

The elements of the physical infrastructure, for example, a hot pipe in the district heating network, are represented by physical connections, which are specialisations of physical edges. Similarly, material (water, gas) or immaterial (energy, data) flowing from one node to another is represented by physical flows. The Technology class, which is a subclass of physical node, is very relevant to the district energy systems as this class represents the various types of energy systems – energy using systems, energy source systems, and storage systems, as shown in Figure 23 below. The different characteristics of the nodes and edges can be represented by the properties' classes and their specialisations. Abstract classes such as DesignProperties, EconomicProperties or PhysicalProperties make this possible.

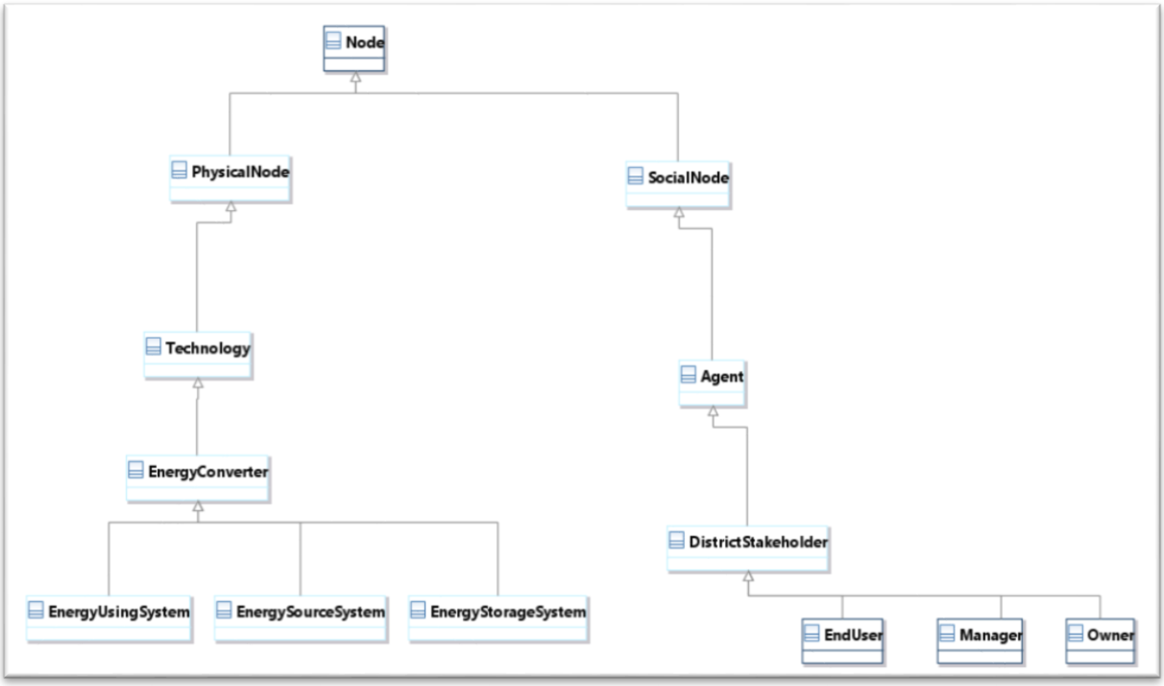



**Figure 23. The link between socio-technical system ontology and `ee-district` ontology concepts (Cardiff University 2014)**

2. Non-ontological resources used in the meta-model

The Resilient project being diverse in content and format meant also reusing and re-engineering non-ontological resources while building the ontology network as done previously in (Suárez-Figueroa et al. 2008). The authors of the deliverable of this European Project 'Neon' put forward best practices and recommendations for transforming resource content (dictionaries, and terminologies) into ontological schemas. Some examples of such resources are normative documents regarding international and European standards. The **`ee-district`** meta-model development followed these best practices to include some of the criteria. For example, the IEC/EN 61970-301 standard is essential to the meta-model to help facilitate the integration of Energy Management System (EMS) applications developed by different entities (IEC61970-301:2011) (McMorran 2007). Doing this enables the Resilient solution (the district coordination system) to communicate with the various energy management systems in individual buildings and energy generation units. Integrating this standard also means that the support of building automation and communication protocols, such as BacNET (ISO 16484-5:2012), is made simpler.

The meta-model linking ee-district ontology adopts a modular architecture and Figure 24 below shows the structure of the modular architecture. The structure shown below is derived from the OntoCAPE's meta-model (Morbach et al. 2007). OntoCAPE is a domain ontology developed for computer-aided process engineering. Although the meta-model was designed for this ontology, it has been used successfully for the development of other domains.

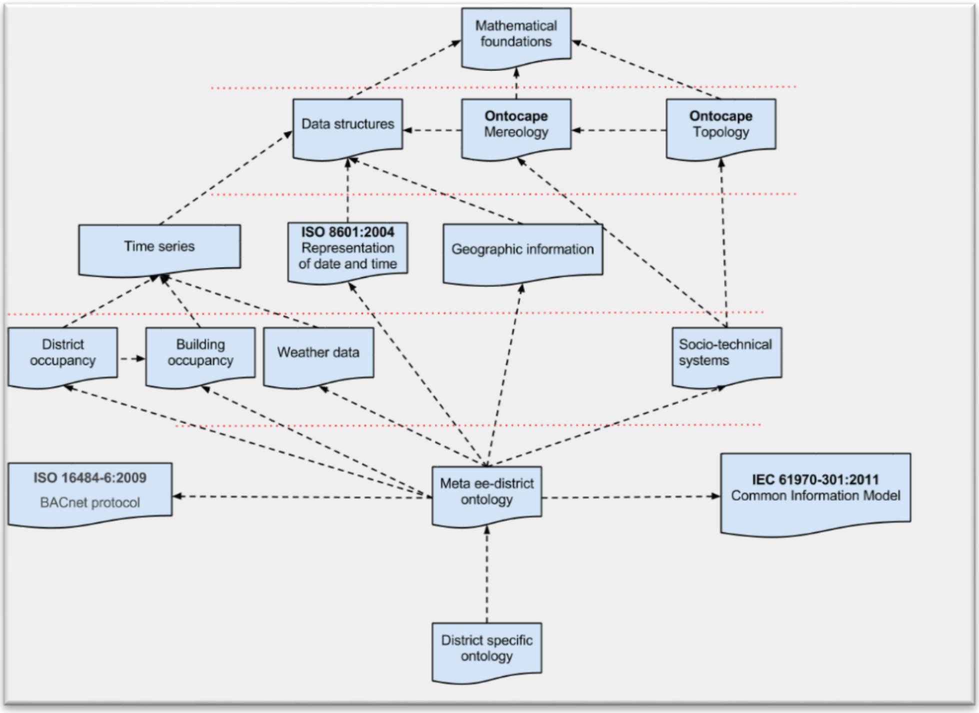



**Figure 24. Module hierarchy of the `ee-district` ontology (Cardiff University 2014)**

*"The ee-district meta-model approach is similar. It formalises a template that encompasses the domain ontologies required to support the definition of an ontology that would be specific to a district."*(Hippolyte et al. 2014). Figure 24 above shows some of the main modules identified, during stage 1 and stage 2 of the methodology, (from Figure 19) that would be part of **`ee-district`** meta-model. Dependency relations between modules are shown using dashed arrows. The socio-technical system ontology has been aligned with the OntoCAPE meta-model, in particular with the topology module (defining fundamental concepts from the theory of connectedness) and mereology module (defining fundamental concepts from the theory of part-whole relations), which are both essential in the standardisation of network and ownership concepts across the ee-district meta-model (Hippolyte et al. 2014).

The **`ee-district`** ontology is aligned with the other domain independent ontologies of OntoCAPE by interposing the generic socio-technical system ontology, as shown below in Figure 25:

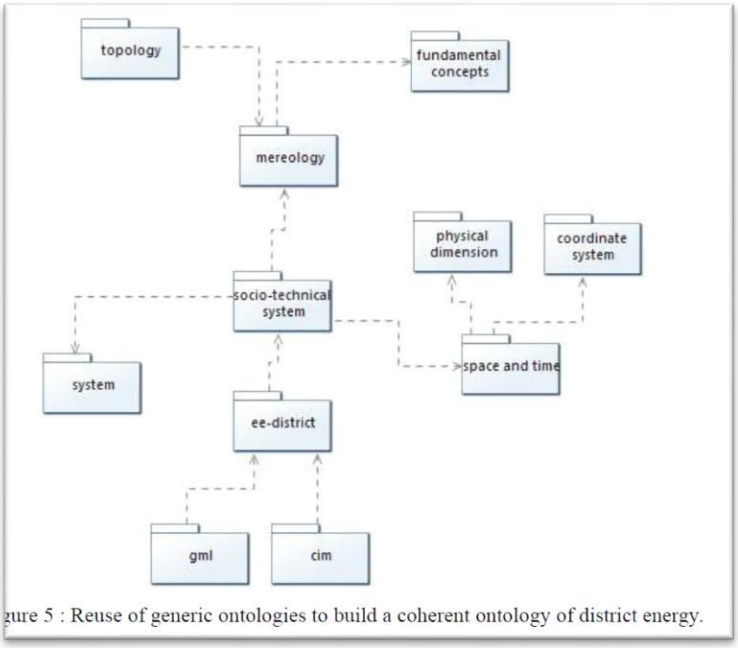



**Figure 25. `ee-district` ontology alignment with other ontologies (Cardiff University 2014)**

All these ontology modules, along with the target ontology, constitute the ontological definition of the ee-district meta-model. Six of these ontologies were reused from the meta-model, which is a modularised ontology design which can be used to support the design of domain ontologies such as OntoCAPE (Morbach et al. 2007) -

a) Fundamental concepts module: the root terms in socio-technical systems and ee-district ontology can be derived from the fundamental classes and relations which are found in this ontology.

b) Mereology module: the part-whole relationships of two types are derived here – aggregation and composition.

c) Topology module: the connectivity between models is made possible with this module.

d) System module: the kinds of systems, their taxonomical semantics and their conceptualisations can be defined through this module.

e) Space and Time module: a specific coordinate system for spatial and temporal data can be defined through this.

f) Physical dimension module: physical quantities, dimensions and units are represented by this module.

The modules are described in detail in the project deliverable (Cardiff University 2014).

*Implementation of the ontology*

**ee-district** ontology here provides high level semantic representation of the district energy system component and organisation which is to be used by any district scale energy management software application purposes (visualisation / monitoring/ generation or distribution control etc.). A web service, developed by author's colleague, Dr. Jean-Laurent Hippolyte was used to encapsulate the ontology which further facilities the knowledge requirement of any energy management software components. The web service also has a SPARQL engine which generates the required SPARQL queries to retrieve knowledge from the ontology repository. Detailed web implementation plan is discussed in project deliverable (Cardiff University 2014).

### 4.2.2. Numerical optimisation for supply-side.

The demand side optimisation and its methodology have already been described and discussed in Section 4.1 through the action research conducted in SportE2. A similar optimisation methodology had to be developed and tested at the supply-side as well. Hence, multi-objective optimisation algorithms on numerical models need to be used at the district level. Doing this would make **REMO** ontology applications consistent at a building and district level in supporting optimisation models and numerical models. The author therefore further aimed to develop:

1. A district energy analytical or simulation model, and
2. An optimisation model which uses this district energy model (from step 1) as the cost function.

This section explains the mathematical (analytical) model and the optimisation algorithm implemented at the district level along with various objectives, decision variables, and constraints of the optimisation problem. The analytical model was built based on the Ebbw Vale site (Resilient pilot site), since most of the site information was available to the author. A more generic mathematical model needs to be developed in the future after considering a variety of district sites which may include various other types of energy sources

*Ebbw Vale site information (Jayan et al. 2016)*


**Figure 26. Ebbw Vale site (Jayan et al. 2016)**

The Ebbw Vale site was previously occupied by a steelworks, which closed down in 1982. The site was demolished after that and remediation was undertaken to develop the site for residential, commercial and educational developments.

The Blaenau Gwent County Borough Council (BGCBC) owns the whole of 'The Works' site, which has six buildings/structures and forms the Resilient project. All of the buildings are located at the northern end of the site, and all except one have been developed within the last seven years. The six buildings/structures are: 1. General Office, 2. Learning Zone, 3. Energy Centre, 4. Multi-storey car park (MCSP), 5. School for aged11-16 pupils, 6. Leisure Centre.

The energy sources in the Energy Centre provide district heating to the entire site. The School, Leisure Centre, and the General Office electrical demand are met by an 8MW HV electricity main which runs through the site. In the case of the other buildings on the site, the main ring supplements the supply from the gas-fired CHP unit. The CHP works in conjunction with the four gas-fired boilers. The boilers are used in series based on the heat demand to provide the district heating system. The council recently also installed biomass boilers in the Energy Centre. The Energy Centre provides district heating to the site, whereas electricity is provided by both the main grid and a CHP unit.

*Thermal Power Supply*

Four gas-fired boilers (ICI REX180 1950kW (input)) and a Cogenco (2012) 375kW CHP plant are used to provide the district heating. The boilers are fitted with Nuway MGN2800 burners with 790-2800kW output, and each boiler has two Variable Speed Driven (VSD) circulating pumps, each rated at 7.5kW and run on duty when there is a boiler demand.

The base load of the district is maintained by the CHP plant and the boilers, which come online as and when needed, are separately metered and connected to the external natural gas grid. Within the Energy Centre, BGCBC owns, manages and operates all of the equipment. The CHP gives priority to meeting heat demand. When the electricity produced by the CHP is not enough to meet the demand at any particular time, it is bought from the national grid, which is quite a common strategy (Liu et al. 2014).

*Electricity Supply*

Both the CHP plant and the main electrical grid combine to provide electricity to the site. The Learning Zone building contains the BGCBC switchboard to which the CHP provides 375kW of electricity. This supply is used by the Leaning Zone, the Energy Centre itself, and the Multi-storey Car Park. The grid directly meets any shortfall in the supply to these buildings. Any surplus production from the CHP goes to the grid and BGCBC receives a small payment under the FIT (Feed-in Tariff) process. On the other hand, the General Office, Leisure Centre, and the aged 11-16 pupils School do not benefit from the CHP or any renewable electricity generation, and are all connected to the main electricity grid.

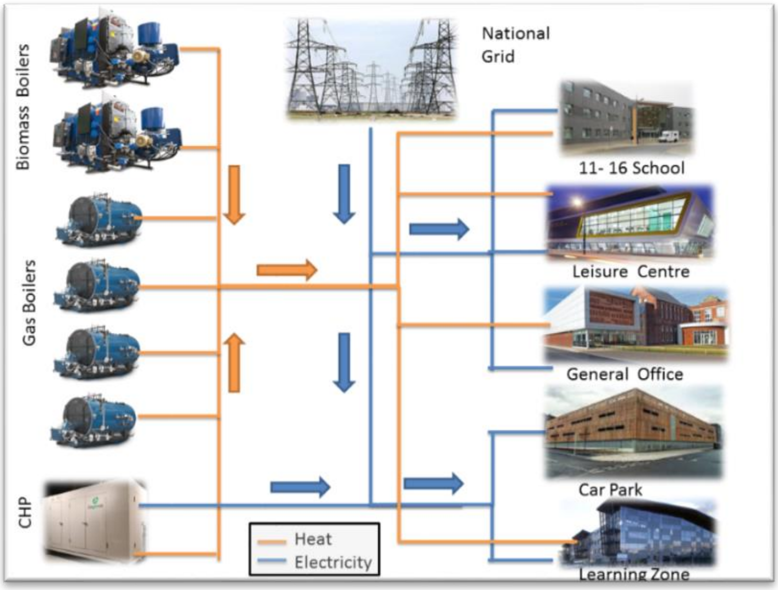



**Figure 27. Heat and power schematic flow in the district (Jayan et. al 2016)**

Currently, the Ebbw Vale uses CHP to meet the base load, and biomass or gas boilers come online when CHP cannot meet the load. Any changes in the strategy of the district operations are made manually by human knowledge, usually based on the seasonal variation. However, the day-to-day energy demand and supply in the district depends a lot on the dynamic and static factors such as weather, occupancy, energy prices and so forth. These need to be taken into account for the day-to-day operational decision-making to potentially increase energy efficiency and also to bring about economic or environmental savings. Optimising the energy generation mix on a day-to-day basis at an operational level, therefore, is a critical issue for the facility managers. Before implementing the multi-objective optimisation, a mathematical model was needed to be used as the cost function of the optimisation. Section 4.2.2.1 below explains the analytical model developed and the reasons for using this model as the cost function in optimisation.

#### *4.2.2.1. Analytical model development*

Development of a district energy model using existing simulation packages (such as Trnsys or EnergyPlus) would compromise on the flexibility regarding what can be considered in the model. For example, considering minute details such as Renewable Heat Incentive (RHI), or carbon tax or emissions due to biomass transport, etc., would not be possible with some of the existing simulation packages. Hence, the decision was made to make an analytical model from scratch using mathematical formulas. The interviews and questionnaires used for ontology development, as mentioned in Section 4.2.1, helped in developing this model.

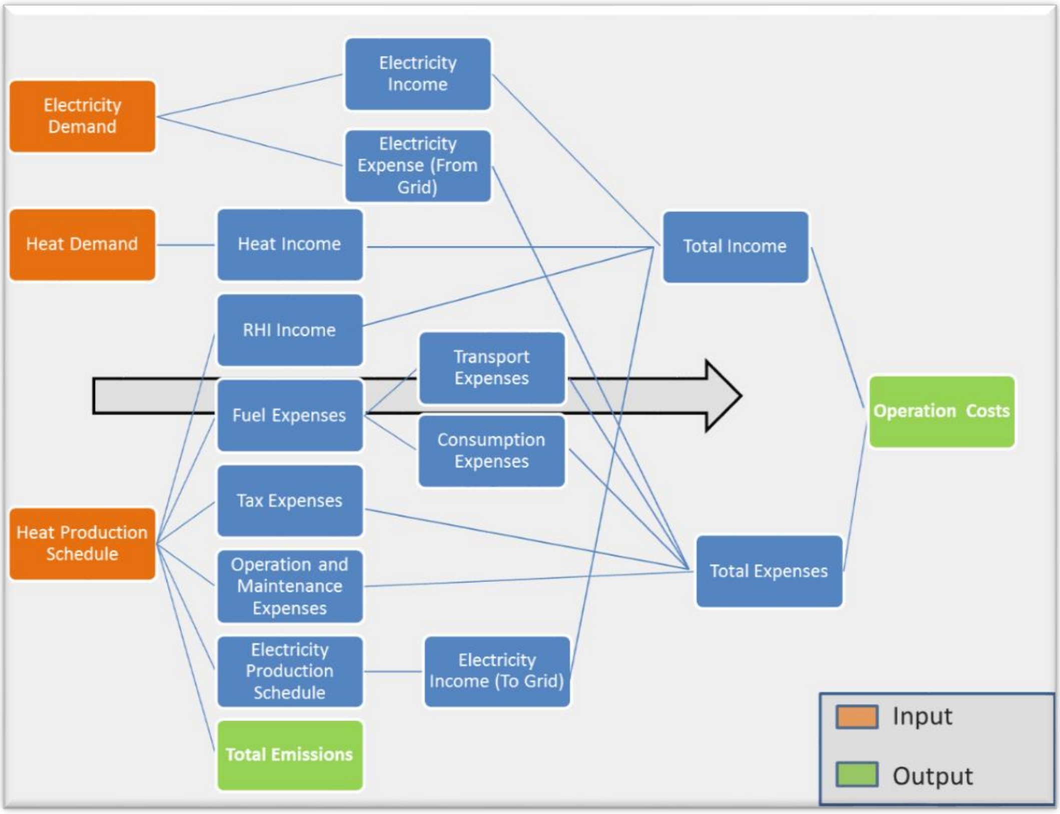



**Figure 28. General mathematical model developed for districts (Jayan et. al 2016)**

Figure 28 above represents a general mathematical model for districts, even though the model is based largely on the Ebbw Vale site. In the future, more sites need to be considered, and newly discovered features or entities need to be included in this generic model.

Here, the electricity demand, heat demand and heat production schedule for a 24-hour period serve as the input. The model is then able to compute the total daily carbon dioxide emissions (kg) and operation costs (£). Emissions can also include other toxic gases such as $NO_X$ and $SO_X$, provided that data is available for this computation. In the case of Ebbw Vale, this data was missing and hence it was not provided.

The model is applied to the Ebbw Vale pilot, and Table 7 below shows the nomenclature adopted in the analytical model.

**Table 7. Nomenclature used in Ebbw Vale Mathematical Model (adopted from Jayan et. al 2016)**

| | Symbol | Representation | Value |
|---|---|---|---|
| Subscript | I | represents the CHP | - |
| | J | represents a boiler | - |
| | K | represents a biomass boiler | - |
| | G | represents an energy source system | - |
| | E | represents the different greenhouse gases | $CO_2$, $SO_2$, $NO_X$, PM |
| Economic parameters | $P^{ELEC}$ | Purchase price of electricity ($£/kWh_{el}$) | Day rate: 0.11<br>Night rate: 0.07 |
| | $P^{GAS}$ | Purchase price of natural gas ($£/kWh_{gas}$) | 0.0248 |
| | $P^{BIO}$ | Purchase price of biomass (£/kg) | 0.205 |
| | $C^{ELEC}_{sale,c}$ | Sale price of electricity (for the energy use system) ($£/kWh_{el}$) | Day rate: 0.11<br>Night rate: 0.07 |
| | $C^{ELEC}_{sale,ng}$ | Sale price of electricity (for the national grid) ($£/kWh_{el}$) | Day rate: 0.0764<br>Night rate:0.03 |
| | $C^{HEAT}_{sale,c}$ | Revenue for delivering heat to the energy use system ($£/kWh_{th}$) | 0.0594 |
| | $C^{Maintenance}_{CHP}$ | Maintenance rate for CHP ($£/kWh_{el}$)) | 0.0035 |
| | $C^{Maintenance}$ | Total maintenance cost for Energy Centre | - |
| | $C^{RHI}_{sale,c}$ | Renewable Heat Incentive for biomass production ($£/kWh_{th}$) | 0.12 |
| Ener gy | $Q^{g}_{max}$ | Maximum production capacities of the energy source system g (kW) | Refer to table 8 |

| | | | |
|---|---|---|---|
| | $Q^g$ | Thermal generation of energy source system g | - |
| | $E^{CHP}$ | Electrical KWh generated by CHP (kWh) | - |
| | $\eta^g$ | Efficiency of the energy source system g (dimensionless) | Refer to table 8 |
| | $*E_{demand}$ | Electrical energy demand (kWh) | - |
| | $E_{sold,c}$ | Sold electricity to the energy using system ($kWh_{el}$) | - |
| | $E_{sold,ng}$ | Sold electricity to the national grid ($kWh_{el}$) | - |
| | $E_{bought}$ | Electricity bought from the national grid ($kW_{el}$) | - |
| | $Q_{sold,c}$ | Thermal energy sold to the Energy using system | - |
| | $Q_{demand}$ | Thermal demand (kWh) | |
| | T | CHP's heat and power ratio (dimensionless) | 0.65 |
| Environmental Parameters | $\mu_e^l$ | Amount of 'e' emitted from the energy source system using fuel 'l' (kg) | - |
| | $\varepsilon_e^l$ | Specific emission of e per kWh for energy source system using fuel 'l' (kg/kWh)) | Refer to table 10 |
| | $\chi^l$ | Calorific value of fuel l (kWh/kg) | - |
| | $\mu_e^{transport}$ | Emissions due to transport of biomass (kg) | - |
| | $BTr_c$ | Carbon emission factor for biomass transport. ($kgCO_2$ /KgBiomass – Km) | 0.00012 |
| Other Parameters | $DIS_{P\&EV}$ | Distance between biomass producer and Ebbw Vale (km) | 277 |
| | $q_{trans}$ | Biomass transported (kg/day) | - |
| | $Cons_{max}^{BIO}$ | Max biomass consumption in one day (kg/day) | 6023 |
| | $Cons^{GAS}$ | Natural gas consumption ($kWh_{gas}$) | - |
| | $N_G$ | Number of energy source systems | CHP, Biomass Boiler 1, Biomass Boiler 2, Gas Boiler 1, Gas Boiler 2, Gas Boiler 3, Gas Boiler 4 |
| | $N_l$ | Number of types of fuel used | Biomass, Natural Gas |
| | $N_{CHP}$ | Number of CHP units | 1 |
| | $N_{BOILER}$ | Number of boilers | 4 |
| | $N_{BIOMASS}$ | Number of biomass boilers | 2 |

* here $E_{demand}$ only considers electricity demand from those customers to which the CHP is linked to.

The numerical values presented here in Tables 7 above and, Tables 8 and 9 below are mostly the data collected from the questionnaire and pilot site interviews (with the owners and facility managers).

**Table 8. Energy source systems' characteristics**

| Energy Source | CHP | Biomass Boiler | Gas Boiler |
|---|---|---|---|
| **Electricity to Heat ratio** | 0.65 | - | - |
| **Max. Thermal Production (kw)** | 410 | 495 | 2800 |
| **Efficiency (%)** | 78 | 82 | 67 |

**Table 9. Parameters related to carbon tax**

| Parameters | Value |
|---|---|
| **CRCTaxRate** | £0.012/kg |
| **CRCElectricityCOnversionrate** | 0.541 |
| **CRCNaturalGasCOnversionRate** | 0.1836 |

*Cost-related Equations*

The operational costs of the district can be calculated using the following equations:

$C_{energy\ centre}$ represents the total cost of the energy centre, whereas $B_{energy\ centre}$ represents the total income received by the energy centre.

The various costs are calculated as shown below:

$$C_{energy\ centre} = C^{BIOMASS} + C^{CONS} + C^{TAXES} + C^{Electricity} + C^{Maintenance} \qquad (1)$$

$C^{TAXES}$ (costs due to taxes), $C^{CONS}$(gas consumption costs) and $C^{BIOMASS}$ (Biomass consumption costs) are calculated as shown below:

$$C^{BIOMASS} = P^{BIO} * q_{trans} \qquad (2)$$

Here, $q_{trans}$ is the quantity of biomass transported and used on site and $P^{BIO}$ is the purchase price per unit biomass fuel.

The biomass transport cost is included in the buying price, according to the pilot owners, and therefore is not explicitly mentioned in equation 2. Biomass storage losses can be ignored as the storage systems on site are highly efficient. Moreover, the biomass pellet fuel is also of good quality, which limits the storage losses. Carbon taxes are calculated as shown below in equation 3.

$$C^{TAXES} = C_{taxes} * (N_{CHP} * \mu_{CO2}^{CHP} + N_{BIOMASS} * \mu_{CO2}^{BIOMASS} + N_{BOILER} * \mu_{CO2}^{BOILER}) \quad (3)$$

$$C^{CONS} = P^{GAS} * Cons^{GAS} \quad (4)$$

$$C^{Electricity} = E_{bought} * P^{ELEC} \quad (5)$$

The cost of maintenance, $C^{Maintenance}$, in the case of Ebbw Vale includes only the CHP maintenance cost, as CHPs need a regular service. This rate is based on the use of CHP and is represented as cost per unit of electrical energy produced (Department of Energy and Climate Change 2008). Investment costs can be ignored throughout the analytical model as they are not part of the district operational stages.

$$C^{Maintenance} = C_{CHP}^{Maintenance} * E^{CHP} \quad (6)$$

The Energy Centre revenue comes from:

(1) The amount of electricity sold to the Learning Zone building,
(2) The excess electricity produced by CHP, during night time, which is sold back to the grid,
(3) The amount of heat energy sold to the Learning Zone building in the district, and
(4) The Renewable Heat Incentive (RHI) received from the amount of biomass produced (Biomass Energy Centre 2011b).

$$B_{energy\ centre} = C_{sale,c}^{ELEC} * E_{sold,c} + C_{sale,ng}^{ELEC} * E_{sold,ng} + C_{sale,c}^{HEAT} * Q_{sold,c} + C_{sale,c}^{RHI} * Q^{BIOMASS} \quad (7)$$

The only building block that consume the electricity which is produced by the CHP are the learning zone and the multi-storey car park. The heat energy consumed by the Learning zone building is the only revenue from the heat energy that the BGCG council receives, as this building is being leased to a third party whereas the council owns all the other buildings.

When considering electricity production, there are two possible scenarios that Ebbw Vale experiences:

(1) The energy centre electricity production is not enough to meet the customer demand, i.e. $E_{sold,c} \leq E_{demand}$ and $E_{sold,ng} = 0$; and

(2) The energy centre electricity production is sufficient to meet all the customer electricity demand, i.e. $E_{sold,c} > E_{demand}$ and $E_{sold,ng} \geq 0$.

The electricity exchanged with the national grid will also affect the costs.

Electricity sold to the national grid:

$$E_{sold,ng} = E^{CHP} - E_{demand} \tag{8}$$

Electricity bought from the national grid:

$$E_{bought} = E_{demand} - E^{CHP} \tag{9}$$

*Emission-related equations*

The equations presented here take into account the $CO_2$ equivalent produced from the various types of fuel used by the energy systems. The Department of Environment, Food & Rural Affairs (DEFRA) provides the greenhouse gas conversion factor for unit energy (Department of Environment, Food & Rural Affairs 2014), as shown below in Table 10.

**Table 10. Carbon Dioxide equivalent**

| Fuel Source | Specific Emissions | |
|---|---|---|
| ***Natural gas (kg/kwh)*** | $\varepsilon_{KgCO2e}^{Natural\ Gas}$ | 0.1850 |
| ***Wood Pellet (kg/kwh)*** | $\varepsilon_{KgCO2e}^{Wood\ Pellet}$ | 0.0118 |

Electricity bought from the grid also has associated greenhouse gas emissions, but this can be ignored as they are part of life cycle emissions.

When source ‘g’ is CHP or boilers:

$$\mu_{KgCO2e}^{Natural\ Gas} = \varepsilon_{KgCO2e}^{Natural\ Gas} * Q^{g} \tag{10}$$

Whereas, when the energy source ‘g’ is biomass boilers:

$$\mu_{KgCO2e}^{Biomass} = \varepsilon_{KgCO2e}^{Biomass} * Q^{Biomass} \tag{11}$$

$\mu_{e}^{transport}$ represents emissions due to biomass transport and this can be calculated as:

$$\mu_{KgCO2}^{transport} = BTr_{c} * q_{trans} * DIS_{P\&EV} \tag{12}$$

$$q_{trans} = \frac{Q^{Biomass}}{\chi^{BIO} * \eta^{BIO}} \tag{13}$$

In equation 12 above, $DIS_{P\&EV} = 276\ km$, because the nearest biomass producer, according to the council, is PBE Fuels and they are 138km away from the site.

$BTr_c = \ 0.00012\ kgCO2\ /KgBiomass - Km$

$BTr_c$ represents the carbon emission factor. This is the rate of kilograms of $CO_2$ emitted by the lorry based on the distance and amount of biomass transported (Biomass Energy Centre 2011a). This gives an approximate value for the amount of $CO_2$ emitted for transporting biomass by road. Assuming that only $CO_2$ emissions are considered for transport, the total carbon dioxide equivalent remains the same.

Consequently, the total greenhouse gas emissions can be calculated as shown below:

$$GHG_{emission} = \left(\sum_{l=1}^{N_l} \mu_{KgCO2e}^{l}\right) + \ \mu_{KgCO2}^{transport} \qquad (14)$$

As mentioned before, the analytical model is also capable of computing other greenhouse gases produced in the process (methane, sulphur oxides, nitrous oxides and so forth). Considering these gases can help take into account the toxicity effect related to solutions, as some of these greenhouse gases are more toxic than others. However, in this work, only $CO_2$ is taken into consideration.

*<u>Other calculations</u>*

This section presents all the other operating equations. The production of thermal energy for gas boilers, CHP, and biomass boilers is calculated below:

$$Q^{BOILER} = \sum_{j=1}^{N_{BOILER}} Q_j \qquad (15)$$

$$Q^{CHP} = \sum_{i=1}^{N_{CHP}} Q_i \qquad (16)$$

$$Q^{BIOMASS} = \sum_{k=1}^{N_{BIOMASS}} Q_k \qquad (17)$$

The electrical energy produced by the CHP can be calculated as shown below:

$$E^{CHP} = \sum_{i=1}^{N_{CHP}} E_i \qquad (18)$$

$E_i$ in equation 18 above can be calculated using the CHP to heat ratio, τ :

$$E_i = Q_i * \tau \qquad (19)$$

Here, $\tau = 0.65$. This is obtained from the manufacturer’s documents.

Assuming the biomass boilers are needed for 24 hours on full capacity, the maximum biomass consumption for one day can be calculated in kilograms, as shown below:

$$Cons_{max}^{BIO} = \frac{N_{BIOMASS} * Q_{max}^{Biomass} * 0.001}{\chi^{BIO} * \eta^{BIO}} \qquad (20)$$

Here, $\chi^{BIO} = 4.8\ kWh/kg$

$\eta^{BIO} = 0.82$

$Q_{max}^{Biomass} = 495\ kW \times 24 = 11{,}880\ kWh$

The maximum biomass consumption calculated for one day is about 6036 kg. The natural gas consumption, which is represented in equation 4, can be calculated as below:

$$Cons^{GAS} = \sum_{g=1}^{N_G - N_{BIOMASS}} \left(\frac{E^g + Q^g}{\eta^g}\right) \qquad (21)$$

#### 4.2.2.2. Supply side optimisation problem

This analytical model can be used as a cost function for optimisation, using a workflow that was greatly used for the SportE2 project, as shown below in Figure 29.

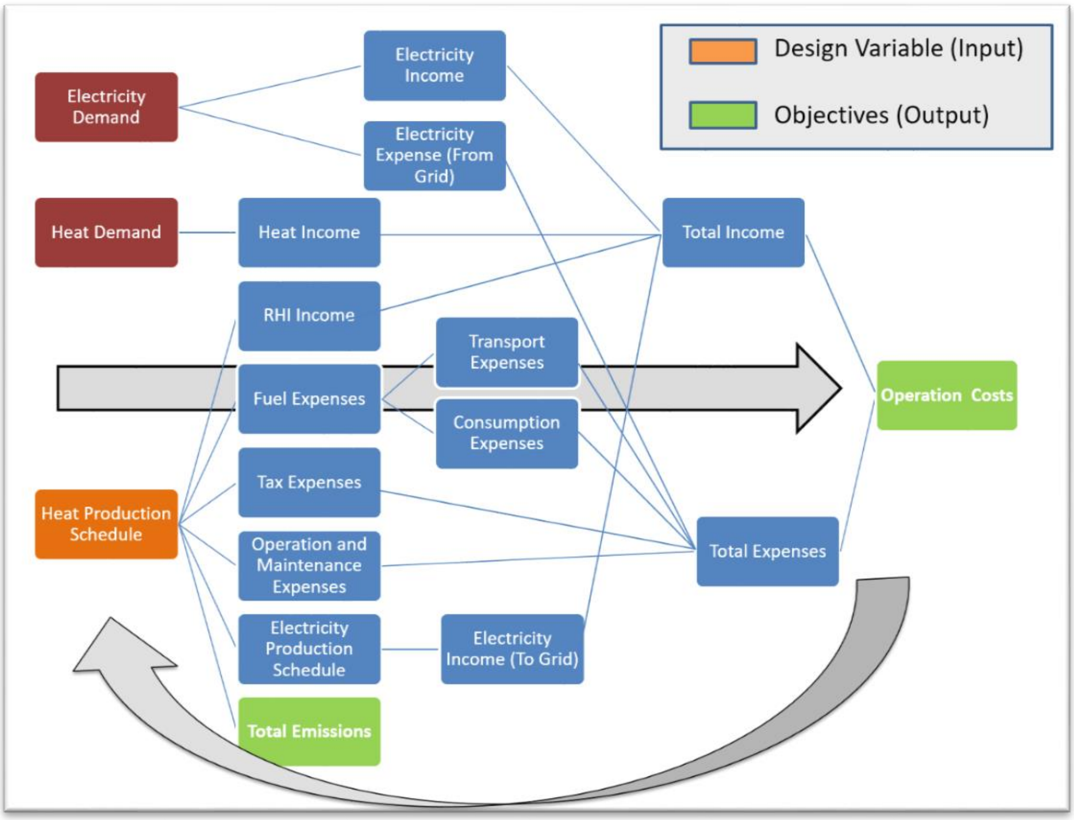



**Figure 29. Optimisation Model**

The optimisation problem can look into solving a multi-objective problem such as minimising costs and minimising emissions while increasing efficiency. The approach here considers emissions, costs and energy efficiency (intelligent matching of supply and demand), and therefore it can be termed 'holistic'. Moreover, the model considers both heat and electricity domains, even though it optimises only heat energy production. The optimisation proposed help facility managers decide on the operational strategy for the district for a period of 24 hours, further helping to optimise the power output of each source in the strategy. The constraints are implemented to make sure that the solution also meets the demand at all times during the 24-hour period analysed. This analytical model is applied to the case of Ebbw Vale district and is used as the cost function for multi-objective optimisation.

*Optimisation Objectives*

- The first objective is related to minimising the operational costs of the district:

$$\text{Minimise}\ \ (C_{energy\ centre} - \mathrm{B}_{energy\ centre})$$

  Detailed calculations for $C_{energy\ centre}$ and $\mathrm{B}_{energy\ centre}$ were presented earlier in Section 4.2.2.1.

- The second objective aims to minimise the total greenhouse gases emissions:

$$\text{Minimise}\ (GHG_{emission})$$

  Calculation of $GHG_{emission}$ was detailed in Section 4.2.2.1.

*Decision Variables*

The 24-hour production schedules in the analytical model for each source $g$, $Q^g$ are chosen as the decision variables. "*According to the IEEE/IEC 61970-301 CIM (Common Information Model) standard, a regular schedule for automation systems can be defined by:*

- *Δt a time step (a constant value in seconds);*
- $t_s$ *a start date;*
- $t_e$ *an end date;*
- $an\ ordered\ list\ of\ time\ points.$

*In the proposed mathematical model, each thermal energy production schedule is represented by a row vector. Let $N_G$ be the total number of energy source systems and energy using systems in the district. $\forall g \in \{1, \dots, N_G\}$, $Q^g \in \mathbb{R}^m$, where m is the number of time points $m = \left|\frac{t_e - t_s}{\Delta t}\right|$. Let $g \in \{1, \dots, N_G\}$, the schedule of energy source system $g$ be denoted by: $v^g = \{v_t^g : t \in T\}$, where T is the set of time points ($|T| = m$). For example, $Q^{CHP}$ denotes the production schedule of the CHP generation system of the considered district.*" (Jayan et al. 2016, p.161).

<u>Constraints</u>

- The difference between the heat production schedule, $Q^g$, and the heat demand schedule, $Q^b_{demand}$, is computed from the analytical model. Here, a factor for losses, i.e. a safety factor, needs to be taken into account as well. The safety factor in the Ebbw Vale problem is assumed to be around 20%, which is assumed by taking an average from the losses seen in historical data. The constraint enables the optimisation algorithm to choose a solution that meets demand at all times.

$$Q^{in}_{ring} \geq \sum_{b=1}^{N_b} Q^b_{demand} \tag{22}$$

- The energy sources have their own maximum and minimum power capacities. These are designed as constraints in the optimisation problem by setting them as lower and upper bounds of the design variable itself. The lower and upper bounds are presented below in Table 11. For CHP and Biomass boilers, the optimisation algorithm however is allowed to turn them off when not needed.

**Table 11. Operational constraints of the problem**

| | Lower Bound (Kw) | Upper Bound (Kw) |
|---|---|---|
| CHP | 375 | 401 |
| Biomass Boilers | 124 | 495 |
| Gas Boilers | 0 | 1600 |

<u>Implementation of district analytical model and optimisation</u>

GA are nature-inspired stochastic optimisation algorithms which have the following characteristics:

- **Encoding**: this is where the decision variables in the optimisation problem are encoded in abstract constructs.

- **Set-based:** a set of abstract constructs (called solutions) are manipulated by the algorithm simultaneously;
- **Iterative:** along the run of the optimisation algorithm the solution set is updated dynamically. This update is performed by applying genetic operators – crossover and mutation operators. The crossover operator combines two or more existing solutions to create a solution, whereas the mutation operator slightly modifies the existing solution by a fraction.
- **Selection and replacement**: the solution set is improved by iteratively exploring the neighbourhood in the solutions search space (defined by the genetic operators). The better solutions discovered are then selected and incorporated.
- **Random-based**: most of the sub-processes mentioned above are applied in a probabilistic manner by the GA.
- **Black box**: the value of a comparative performance measure for each solution (called evaluation) is then needed by GA, regardless of how this measure is computed.

Here, the NSGA-II algorithm developed by Prof. Kalyanmoy Deb was used (Deb et al. 2002) and implemented in MATLAB[13].

[13] http://www.mathworks.com/matlabcentral/fileexchange/10429-nsga-ii--a-multi-objective-optimization-algorithm

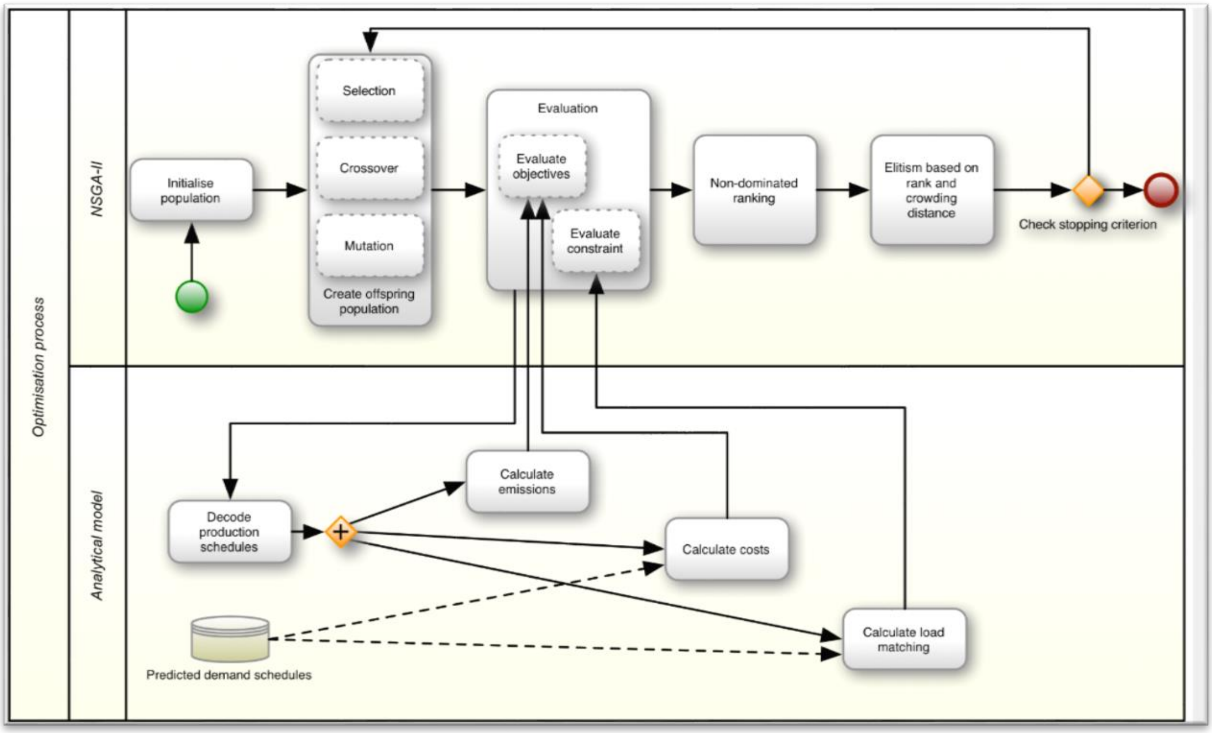



**Figure 30. Optimisation process workflow showing the use of NSGA-II and district analytical model (Jayan et al. 2016)**

The overall optimisation process workflow and the interaction between the analytical model and the NSGA-II optimisation algorithm is shown above in Figure 30.

A straightforward generalisation of a classical single-point crossover operator is chosen. This operation means that the row vectors in the schedule matrix (decision variable) are crossed by randomly picking a time point from either side. The scope of this research does not demand a detailed look at the optimisation model parameters and therefore the selection of mutation operator is also basic. *"When mutation probabilistically occurs, a random value of thermal energy production is given at a randomly picked time point of a randomly picked schedule (respecting the lower and upper bound capacities of the targeted generation system)"* (Jayan et al. 2016, p.162). Non-dominated sorting is performed within the algorithm by evaluating the constraints of the optimisation problem. The NSGA-II algorithm used maximum generation, in this case, 100, as the stopping criterion for the optimisation problem.

As mentioned earlier, the decision variable solutions are encoded as $m$-by-$N_G$ matrices. Conceptually, they are matrices of regular schedules as standardised in IEEE 61970-301 (British Standards 2011). The results of the optimisation problem are presented in the next section.

#### *4.2.2.3. Results and analysis of supply side optimisation*

Two different optimisation test cases were applied to the Ebbw Vale pilot using typical winter day demand profiles, and the results of these were used to compare with the base case (which is also known as the business as usual case). The testing results and analysis are shown below:

**Business as usual case** – the business as usual case utilises simple deterministic algorithms to calculate the production schedules in the district and is currently implemented in the pilot site. The operational strategy is manual (under the control of the facility manager), and it uses the CHP initially to meet the base load, and any excess load is met by the biomass and gas boilers. The CHP is given priority to meet the loads. The biomass boilers are preferred over gas boilers for three reasons:

(1) The carbon emissions produced using this source is almost 10 times less compared to gas boilers,

(2) Biomass boilers are capable of running at a lower output power, which can reduce the excess energy being produced, and

(3) They are economically attractive, as renewable heat incentives (RHIs) are available for unit heat production using biomass boilers.

Figure 31 below shows the results where each source is used to meet the demand. The deterministic algorithm uses a mathematical rule-based approach to computing the production profiles, which were generated based on human knowledge of system running. Figure 31 shows demand being perfectly met by production profiles with the available resources. Table 12 shows the final cost and emissions value calculated during the 24 hours.

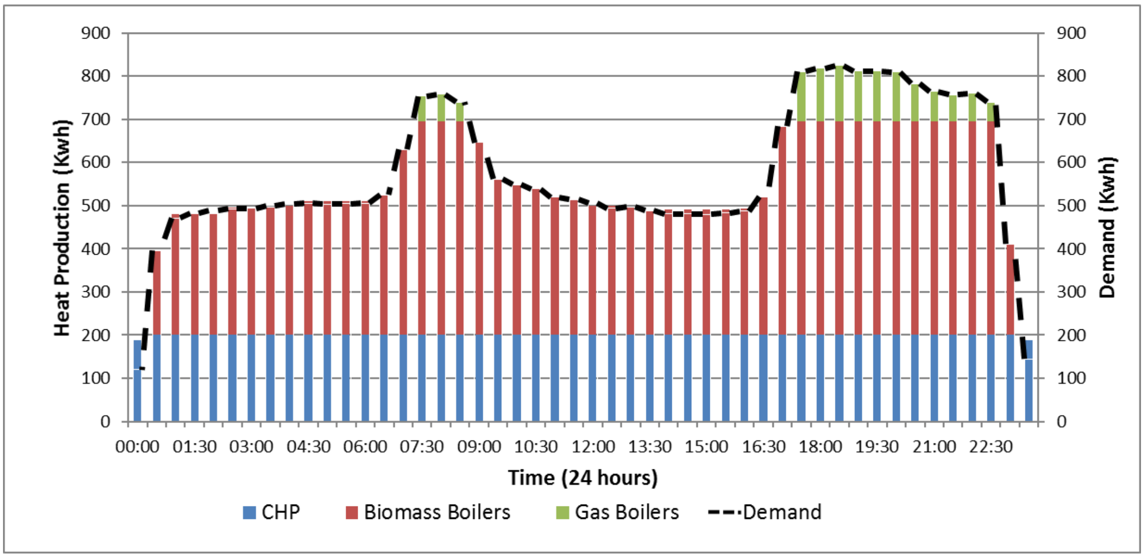



**Figure 31. Energy Demand vs. Supply for business as usual case.**

## Scenario 1: Optimisation using NSGA-II

In this scenario, the NSGA-II algorithm was used to solve the problem. NSGA-II also has user-defined parameters, like other GAs, which impact the performance of the algorithm. During the running of the algorithm, sufficient variety among the population sets needed to be induced and therefore the algorithm was assigned a crossover value of 0.9 and population size of 1000, considering the scale of the solutions. 0.05 was taken as the mutation value, making sure that the solution would not converge early.

The strategy regarding the order of use of sources was fixed and therefore the optimisation could only modify the output power of each of the sources within its constraints. One of the non-dominated solutions achieved by the NSGA-II algorithm is shown below in Figure 32. The algorithm is adapted to the energy management problem, and it processes the decision variables before the evaluation stage to make sure that the demand is met for each time step. The cost and emissions value achieved through this scenario is shown in Table 12.

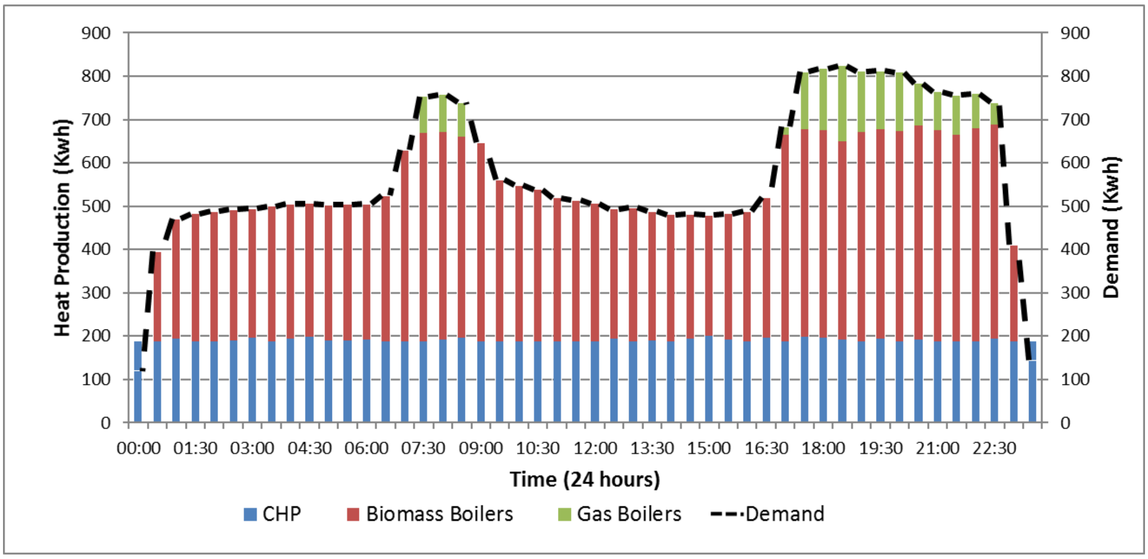



**Figure 32. Energy Demand vs. Supply for scenario 2 using NSGA-II.**

### Scenario 2: Optimisation using NSGA-II but with a changed strategy

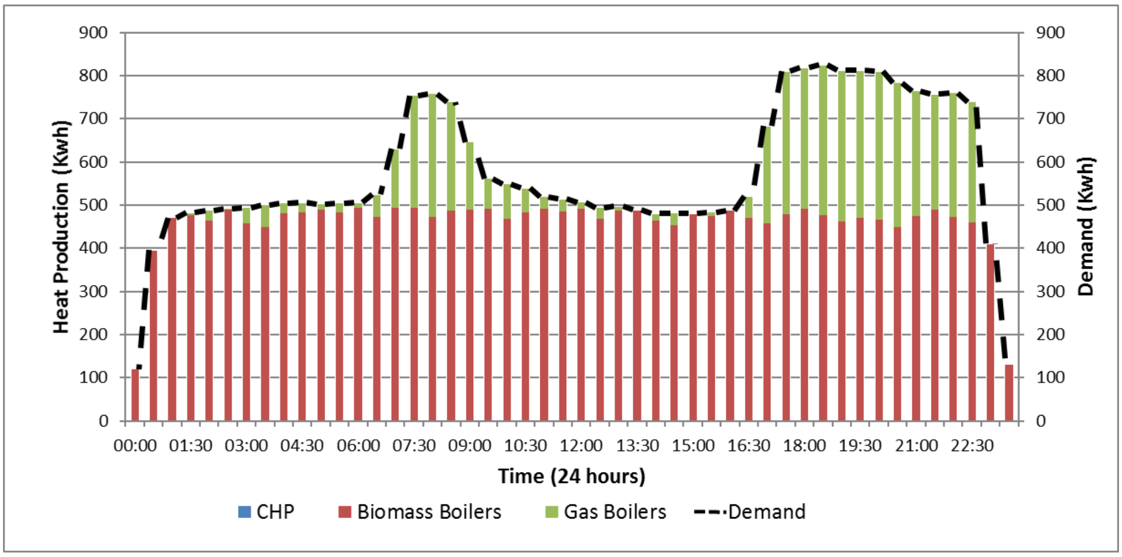



**Figure 33. Supply vs. Demand for Scenario 2**

Scenario 1 did not make a huge amount of savings in terms of profit and brought about a very small decrease in emissions (results are shown in Table 12 below). In other words, very few solutions could meet all constraints while keeping emissions and costs low. In Scenario 2, therefore, the algorithm was made flexible regarding the strategy, i.e. change the order of use of the energy sources. Any change in strategy was performed manually by testing out many different possibilities. Doing this brought about a drastic improvement in the cost and emissions objectives. Figure 33 above shows one of the non-dominated solutions where biomass boilers and gas boilers were used, and CHP was turned off. The objective values computed for this solution are shown below in Table 12.

The loss in the network, which is approximated at around 20%, is included in the demand profiles of Figures 31, 32, and 33. This value was approximated by historical data analysis. The final results of the objectives for each scenario and their computational time, during a typical winter day, are shown below in Table 12:

**Table 12. Final results (Jayan et. al 2016)**

| | **Business as Usual** | **Scenario 1- Optimisation using NSGA-II** | **Scenario 2- Optimisation using NSGA-II (flexible strategy)** |
|---|---|---|---|
| Computational time (seconds) | *1.98* | *2373* | *2379* |
| Profit (£) | £1854 | £1805.18 | £2442.68 |
| $CO_2$ Emissions (kg) | 2394.7 | 2374.89 | **1531.92** |
| No. of time steps where Demand was not met | 0 | 0 | 0 |

*Discussion*

When compared to the business as usual case, scenario one reduces emissions by 0.8% during the 24-hour period analysed. Despite the NSGA-II optimisation being applied, this is very little improvement in results. The optimisation also failed to improve the profits, which actually decreased by 2.6%. When compared to the deterministic algorithms used in the business as usual scenario, scenario one has not brought about any drastic improvements. Therefore, for scenario two the strategy that decides the order of priority of use of the energy sources was made flexible, which improved the results – it increased the profits by 31.8% and most importantly, decreased the emissions by 36% when compared to the business as usual case.

The results are purely computational and not monitored in reality. However, the optimisation model does consider operational constraints, through sub procedures applied to each individual solution of the population, before calculating the objective values such as profits and emission. One of the constraints for example is the fact that CHP should not be switched on and off repeatedly over a short period of time, as it has a minimum start-up time to work at full capacity. Therefore, solutions which suggests such an operational schedule are corrected through sub procedures prior to the evaluation stage of NSGA-II.

For the current demand profile analysed, turning off the CHP and using biomass boilers and gas boilers worked out to be a better strategy. One of the reasons for this could be the high renewable heat incentive received for every unit of energy produced from the biomass boilers. However, the results might vary with varying different demand profiles. Therefore, the optimisation algorithm should be modified in future wherein by default the strategy is kept flexible, which means it would be capable of autonomously running the different test cases with different strategies rather than leaving it to the user to do it manually. Applying the black box approach to optimisation therefore can be advantageous in such cases.

In conclusion, NSGA-II can bring savings in costs and emissions if it is allowed to be flexible with the strategy and not restricted to finding the optimum output power of energy sources. The advantage of the optimisation performed here is that it takes into account all constraints, factors, and objectives to compute a feasible solution. When considering these factors, the GA can produce an optimised solution which can perform better than the operational strategy devised through human knowledge. The Ebbw Vale site, being small compared to other districts, has fewer feasible strategies for running the district. However, for larger districts with many energy sources, the problem can become complex, and the optimisation methodology adopted here can be beneficial in such cases.

One of the challenges behind the optimisation is the computational time it takes to process results. *"The optimisation process currently is time-consuming as it takes about 2373 seconds hours on a normal computer with which has the following specification- Intel(R) Core (TM) i5-3360M CPU @2.80 GHZ Processor Speed, RAM 8 GB"* (Jayan et al. 2016, p.164). However, if the optimisation is to perform 'day-ahead' scheduling for facility managers, it becomes less of a problem as it gives ample time for the solutions to be implemented. Other options would be to follow a similar approach to that in SportE2 wherein high-performance computing was used and the optimisation process was also parallelised (Petri et al. 2014a).

*Limitations and future work*

For the current analysis, no real-time prediction of building demand is used, and the experiments are carried out offline using a typical winter demand profile for each building, which was provided by project partners. To implement the optimisation methodology, in reality, the 24-hour demand profile predictions of the various buildings that consume energy in the district are needed. These demand loads are dynamic in nature based on factors such as weather, occupancy, seasons, etc., and hence *"being proactive and planning ahead can increase the energy efficiency of the overall system"* (Jayan et al. 2016, p.164). Demand prediction is possible by following a similar approach to that used by the authors in the SportE2 project using artificial neural networks.

Future work can also look into optimising the supply and return temperatures within the network. To do this, however, a dynamic simulation model of the network is needed. Once this is available it can be linked with the analytical model and furthermore the energy production schedules can be translated into the actual setpoint – for example, supply temperature, return temperature, the mass flow of water, etc.

The current mathematical model does not include any renewables and storage technologies because it is largely based on the Ebbw Vale case study where these technologies were lacking, but this needs to be integrated in the future. More case studies in the future can help add knowledge to the existing analytical model, making it more generic. The study conducted here is steady state, which means the time steps are not linked to each other and therefore latency effects and time constants cannot be considered during the analysis.

As mentioned in the discussion earlier, the optimisation code ideally should be altered to allow for NSGA-II to be flexible regarding order of use of the energy sources. Currently, any change in strategy needs to be manually made in the code, which is not feasible when the model needs to be applied to a slightly complicated district that has many energy sources.

In this research, only the NSGA-II algorithm was implemented; further investigation and testing can be done using other GA-based optimisation algorithms for better results. A range of GA operator instances (selection, crossover, mutation) can also be experimented, starting with ones that has previously solved similar real-world optimisation problems and moving towards the ones that are tailored for district energy production scheduling. *"The resulting best-practice metaheuristics set-up for district energy schedule optimisation, combined with accurate load prediction and possibly deployed on delocalised high-performance computing infrastructures, could be at the heart of a multi-criteria"* (Jayan et. al 2016, p.164).

## 4.3. Conclusions from action research

Although the Resilient project looks into real-time district energy optimisation, it fails to consider any demand optimisation within the building domain, which was also a gap highlighted during the literature review in Chapter 2. Similarly, SportE2 fails to consider any supply side optimisation. Therefore, the need to harmonise supply and demand side optimisation was reinforced through working on these two projects, realising the true potential of bringing together both solutions adopted in both the projects. A method to harmonise both demand and supply side optimisation can be concluded as a result of the action research conducted as shown in figure 34 below.

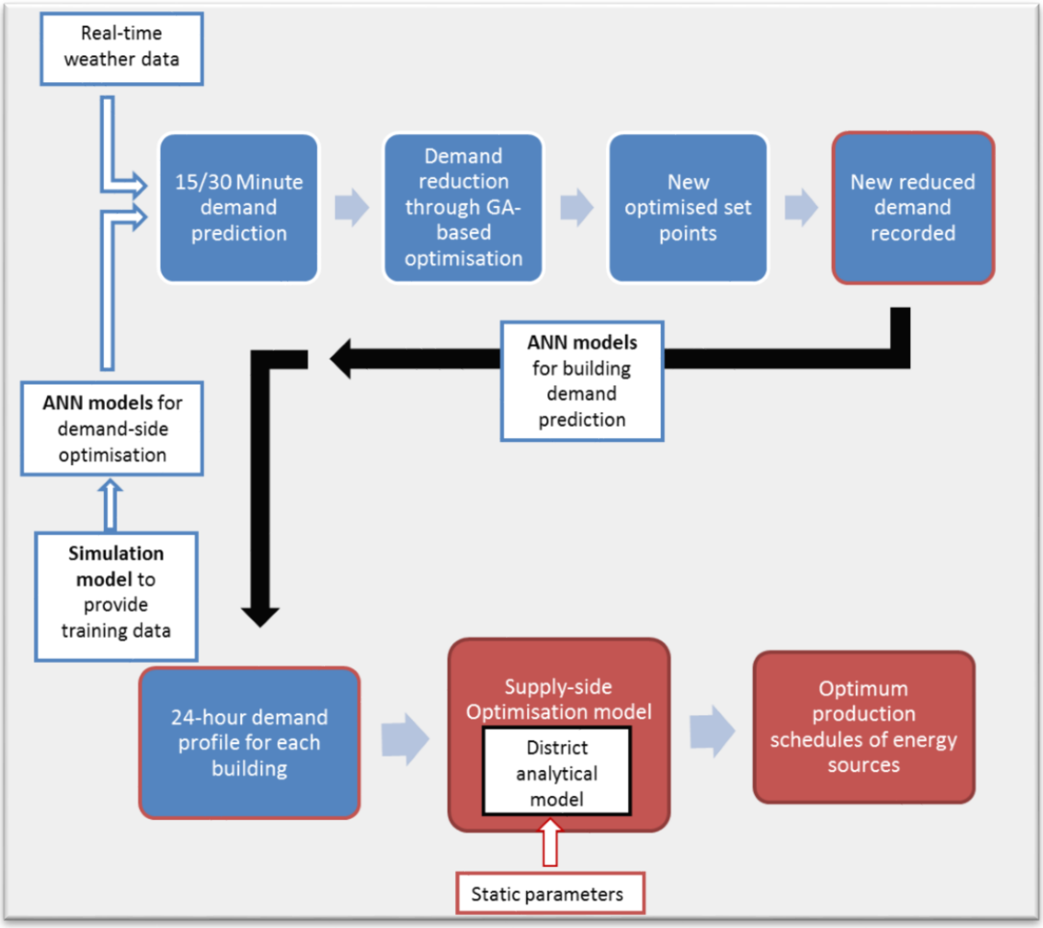



**Figure 34. The link between building energy optimisation and district energy optimisation**

Here, the results of daily demand optimisation (at a building level) would need to be taken into consideration for the supply optimisation (at a district level). The author believes that this can achieve a holistic energy optimisation. The demand side energy optimisation is conducted through the SportE2 solution wherein ANN models are used with multi-objective optimisation algorithms. A pre-requisite for this would be ANN models for each use case/scenario to be trained and validated (either through simulation model or historical data). This solution would optimise the demand throughout the day (every 15 minutes or 30 minutes). The new, reduced demand schedules of each building will be used to predict the 24-hour demand profile for the next day (again using demand prediction ANN models). The predicted demand profile can consequently be used as an input to the simulation/analytical model of the district for supply side optimisation. Hence, the optimisation of the supply side can be linked to the optimisation of the buildings. Together they would deal with the complex issue of trying to match demand and supply in real time taking into account all the static constraints and objectives, and consequently increasing the overall efficiency.

ANN models which are capable of day ahead demand forecasting is needed to complete the workflow. Initial work on this suggested using the ANN input and output as shown below. However, detailed investigation is needed in this area. The author assumes the ANN model shown below for the purpose of completing the ontology in this research.

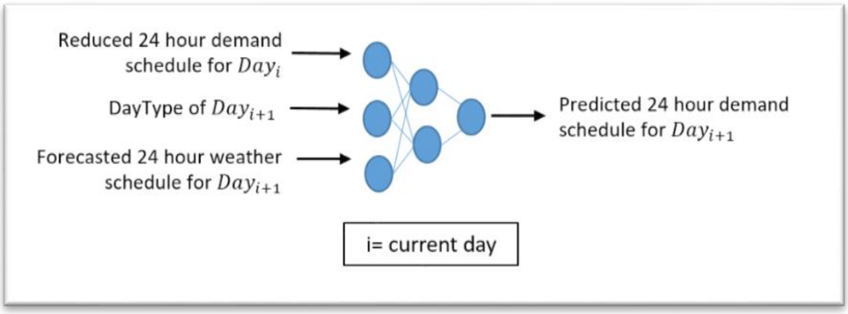



**Figure 35. Proof of concept of ANN model for day ahead demand prediction.**

***REMO** ontology conceptual model design*

**REMO** ontology which needs to be developed would then have to support this harmonisation of demand side optimisation (SportE2 demand optimisation method) and supply side optimisation (district energy model and optimisation method). To do this, the following concepts can be concluded to be part of **REMO** ontology as shown in figure 36 below:

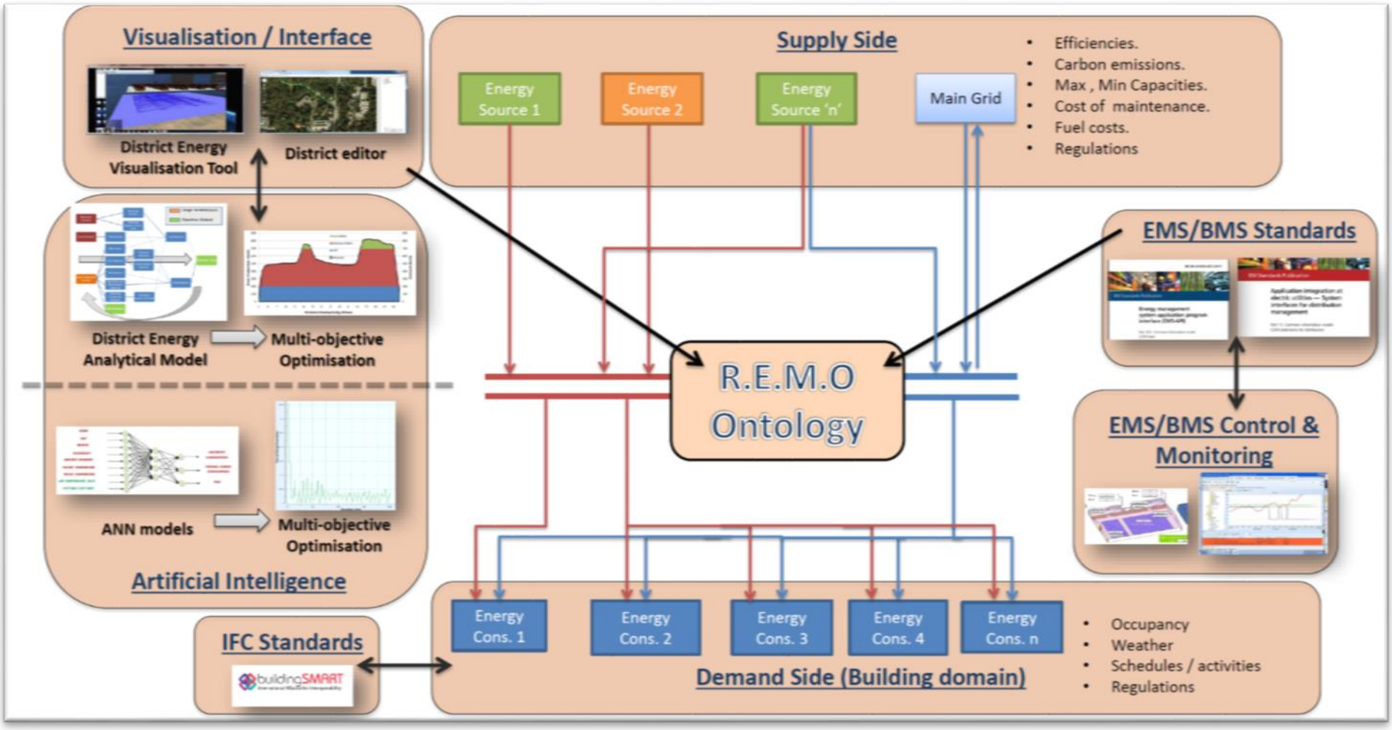



**Figure 36. REMO ontology concepts further detailed from action research**

- **Artificial intelligence concepts** – these include entities that are meant to support the running of AI models such as prediction or optimisation models. AI models as mentioned earlier are key for any real-time energy optimisation.
- **Numerical or simulation model concepts** – simulation or numerical models, as seen in previous works, are also useful to analyse different scenarios in advance and help the decision-making process. The entities identified here would, therefore, support the running of these models.
- **Energy Demand side concepts** – these entities are mainly building or demand side related. Various aspects of a building that are needed for real-time energy management and optimisation are included here.
- **Energy Production side concepts** – these entities are mainly district related or production side related. Various aspects of the district that are needed for real-time energy management and optimisation are included here.
- **Link with IFC and other standards** – the ontology would also need to include some of the entities that were defined previously in standardised ontologies. For example, in the building domain, IFCs are standard data models which might need to be included or linked with **REMO** concepts as well.

- **User interface concepts** – these concepts are needed in the ontology to support any user interface that may be developed as a part of the application through which users can interact with the ontology (or other features of the energy management system) for well-informed decision-making.

Any optimisation based on artificial intelligence needs to involve real-time data to reduce the performance gap, as mentioned earlier in Chapter 1. **REMO** ontology, therefore, would need to be linked with automation systems in buildings (BMS) and districts (EMS) as shown before in figure 36. Consequently, systems or frameworks using this ontology would have the know-how to be able to interact with sensors and actuators in the respective buildings and districts. The main application of **REMO** ontology would be to:

- Represent intelligent energy information (through the user interface).
- Support real-time intelligent energy management by linking artificial intelligence and automation systems.
- Support the optimisation of building energy or demand side: Holistic energy management within buildings – taking into account the various objectives involved.
- Support the optimisation of district energy or supply side: Holistic energy management in districts taking into account environmental and economic objectives.

The research also planned to make the artificial intelligence based use cases adopted in **REMO** ontology replicable to other districts of the future, which are similar to Ebbw Vale (as right now it is built based on this particular site). This can be possible if the contextual understanding of the use case-based dependent and independent governing variables and their complex interactions is captured in **REMO** ontology. The rule axiom features in ontologies are capable of doing this. Consequently, when **REMO** ontology is instantiated for another district in the future, the knowledge required for developing the optimisation and prediction models are available through the reasoning process. This makes **REMO** ontology very unique, innovative and intelligent as it is embedding the action research knowledge of use cases generation in the ontology. Chapter 5 further discusses how the ontology can be part of framework which is capable of facilitating real-time holistic energy management.

# 5. Overall System Design

This chapter explains the overarching system framework which aims to aid real-time energy management by using ontologies to facilitate the working of artificial intelligence solutions and underlying automation systems. In this chapter, Section 5.1 describes the overall framework where the different layers are presented briefly. The core part of the framework is the semantic layer, also the major contribution of this PhD, is discussed in detail in Sections 5.2. Generic use cases of the overall framework are presented in Section 5.3.

## 5.1. Overarching Framework

The design of the overarching framework is shown below in Figure 37.

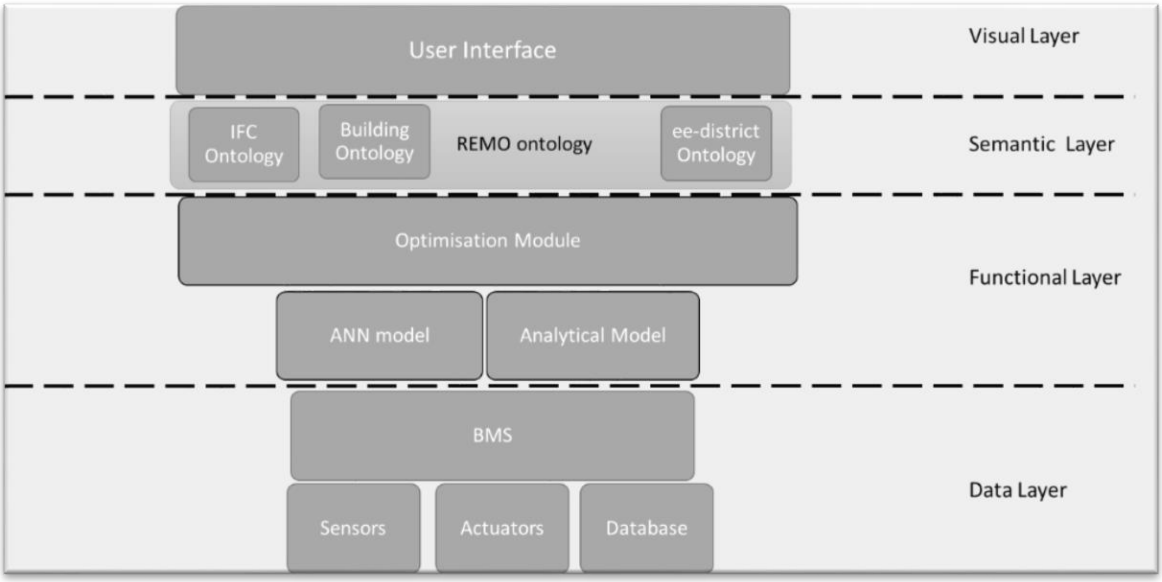



**Figure 37. Architecture of overarching framework**

*Semantic Layer*

The brain of the overall framework is the semantic layer, which embeds the **REMO** ontology at its core. **REMO** ontology includes relevant parameters and variables related to energy management at both the building and district domains, which is capable of supporting the optimisation operations in buildings and their districts using the methodology adopted in SportE2 (Section 4.1) and district energy optimisation (Section 4.2.2) respectively.

**REMO** ontology here imports **ee-district** ontology (from resilient), IFC ontology and a refined version of the building ontology developed by the author under the SportE2 project (Jayan et al. 2014), as shown in Figure 38. Similar classes and properties from these ontologies are aligned to the **REMO** ontology using a linked data approach. The various ontologies within **REMO** are:

- **ee-district ontology** – this ontology was developed as a part of the Resilient project (Section 4.2 in Chapter 4). This ontology is also linked to other existing standards such as the CIM standards, socio-technical ontology, and other well-defined ontologies such as the OntoCAPE ontology. It mainly contains all the relevant concepts with regard to the district and the energy systems in the district.
- **Building ontology** – this ontology consists of the concepts that were developed under the SportE2 project (Section 4.1). These concepts are used for real-time energy management at a building level.
- **ifcOWL ontology** – the idea behind linking IFC domain ontology (**ifcOWL**[14]) with **REMO** ontology was to make sure that the energy management applications through **REMO** would be BIM compliant. The concepts in **ifcOWL** ontology similar to **REMO** ontology are therefore aligned together using equivalent property and rules. This means an instance model of **ifcOWL** can be used to automatically instantiate **REMO** ontology through a reasoning process. This can be useful, as it means that any future building that is available in IFC format can be used to instantiate **REMO** ontology, provided the IFC model is converted into an ontology. The IFC to OWL convertor (Terkaj and Šojić 2015) can be used to convert IFC files to an ifcOWL file. More details on this are provided in Section 5.1.4.

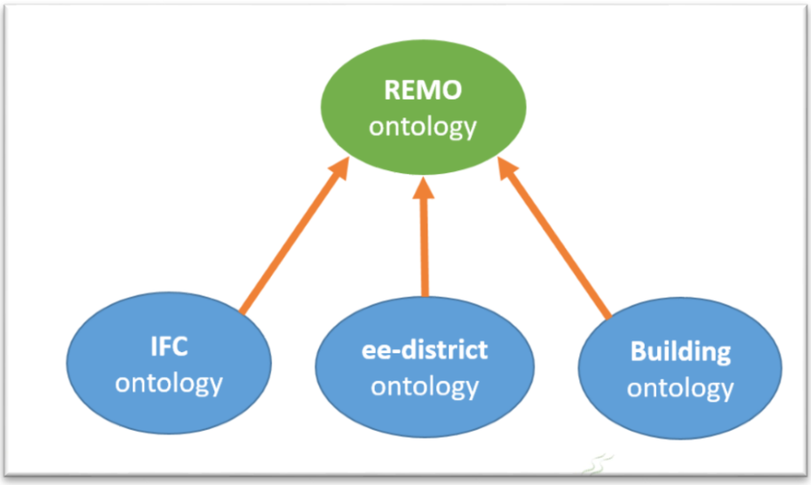



**Figure 38. REMO ontology importing the necessary ontologies**

[14] http://www.buildingsmart-tech.org/future/linked-data/ifcowl

Section 5.2 explains the detailed design of the **REMO** ontology including classes, properties and rules, and also shows links between **`ee-district`** and IFC ontology.

*Functional Layer*

The functional layer includes the prediction models, optimisation models and simulation models which run the optimisation and prediction use cases for real-time energy management in buildings and districts. It is used for:

*Demand side optimisation for the buildings* – this is based on the action research work carried out in the SportE2 project (SportE2 Project Consortium 2014), as explained in Section 4.1. Demand side optimisation functionalities use the ANN models and their respective optimisation algorithms.

*Supply side optimisation for the district* – this functionality allows real-time optimisation of the production side in the district using mathematical models and optimisation algorithms, as discussed previously in Section 4.2.2. The supply side optimisation can take into consideration the optimised building demand on a day-to-day basis, as per the overall workflow demonstrated in Section 4.3.

*Visual Layer*

The framework also has a visualisation layer, which is the user interface. The user interface is envisioned to be linked to the semantic layer where the ontology is hosted. The scope of this research does not include looking into the interaction between the semantic layer and the visual layer; however, preliminary work on this is presented in Chapter 6, under Section 6.3.

Through the visual layer, the users can interact with the overall framework and run the use cases for energy optimisation and management. More details on the development can be found under Section 6.3.

*Data Layer*

The data layer represents the parameters and other information stored in the BMS of buildings and EMS of districts. The BMS and EMS are of particular importance for all the dynamic information within buildings and districts, respectively. The functional layer accesses this information and further uses it for running of optimisation, prediction, and mathematical models.

Detailed interaction between the different layers is beyond the scope of this research. The interaction may require the development of a system integration module which is similar to the approach taken in the European project ISES where the module served to link the data with systems and physical devices in place, using a linked data approach (Katranuschkov et al. 2015). The Semantic layer is discussed in detail below in Section 5.2.

## 5.2. Semantic layer

This section describes the **REMO** ontology in detail and also some of its links with other ontologies.

### 5.2.1. REMO Ontology Classes

Figure 39 below shows the high-level concepts derived as a result of the action research.

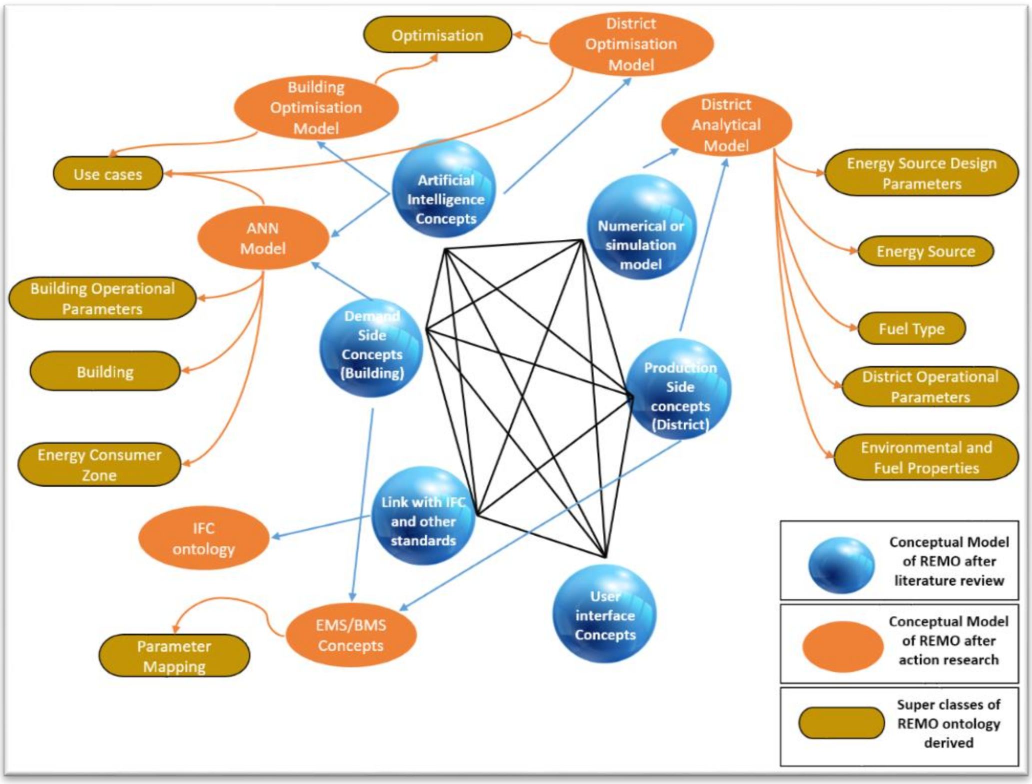



**Figure 39. High-level concepts of REMO ontology derived from action research**

The main classes of **REMO** ontology were derived from the action research conclusions as shown in Figure 39 above. The blue spheres in the diagram show some of the high-level abstract concepts of **REMO** ontology derived post literature review stage. Whereas the orange oval-shaped concepts drive into the details of some of the abstract concepts, clarity on which, was a result of the action research conducted. This can then further be derived into super-classes. The **REMO** ontology super-classes are the direct subclass of `owl:Thing` as shown below in Figure 40.

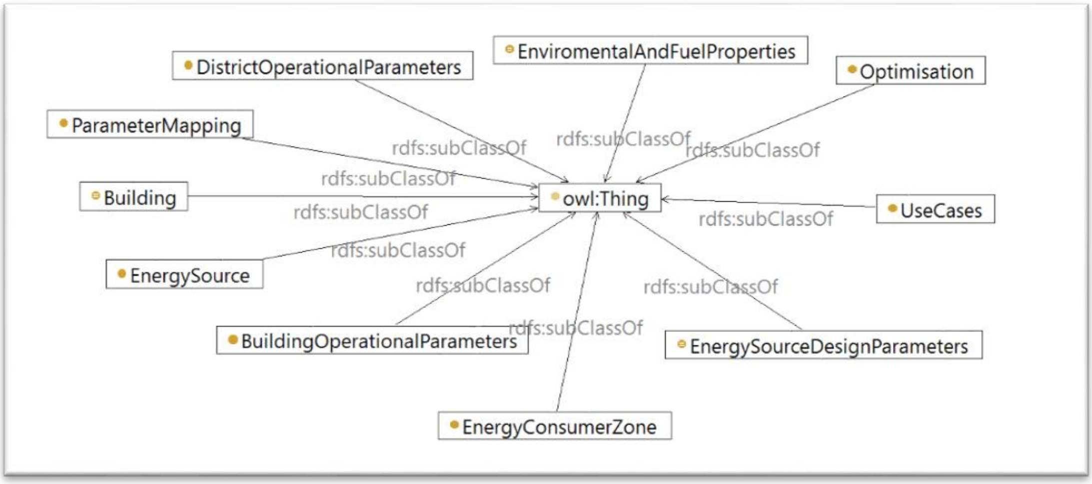



**Figure 40. Main classes of REMO ontology**

This section describes these classes, their subclasses and their purpose in the **REMO** ontology. The classes of **REMO** ontology are grouped according to their domain or application area. Only the main classes are described in this section. However, a description of the other classes can be found in the appendices.

1. *Classes related to building and energy consuming zones*

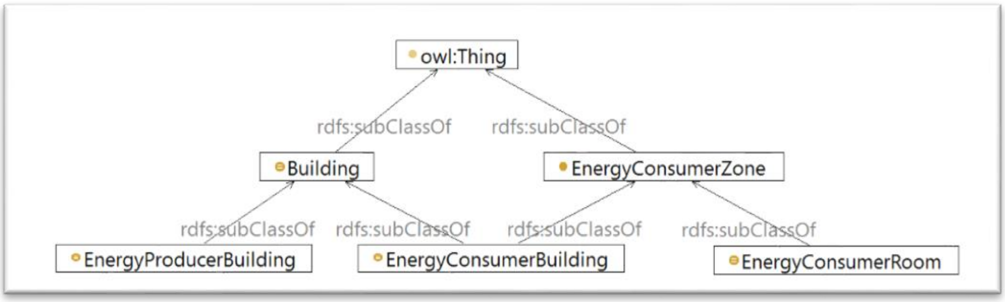



**Figure 41. Building and EnergyConsumerZone class**

### `Building` class

<u>Description</u>

This class contains all individuals representing buildings in the district.

<u>Relations</u>

- `Building` is a subclass of `owl:Thing`

- `Building` class also has an equivalent class linking to **`ee-district`** ontology, which is `eedistrict:Building`. The prefix '`eedistrict`' here suggests that the class belongs to **`ee-district`** ontology.

**`EnergyConsumerZone` class**

Description

This class contains individuals that are zones in buildings. A zone here represents a space that can control its heating and cooling requirements. This space can be one room or a group of rooms. Building energy optimisation use cases that can be applied to individuals of this class.

Relations

- `EnergyConsumerZone` is a subclass of `owl:Thing`.

Usage of Class

The following naming convention is applied to name individuals of this class:
*<name of index class>_<name of energy consumer zone>*.
Example: *EnergyConsumerZone_OfficeSpace1*

2. *Classes related to Energy Sources*

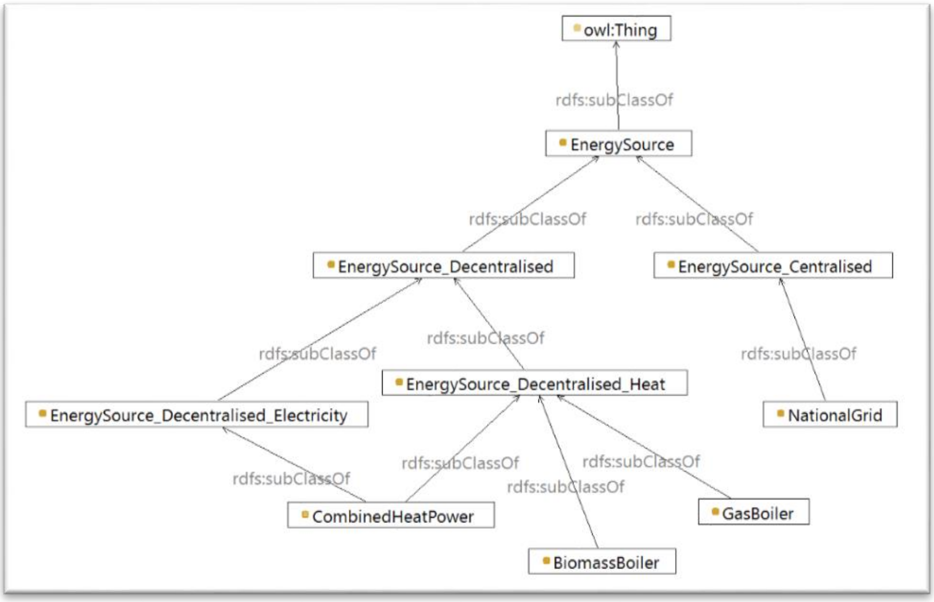



**Figure 42. `EnergySource` class and its subclasses**

**`EnergySource` class**

Description

This class contains all individuals that are energy-producing systems in the district. They can be a central source or a decentralised source.

Relations

- `EnergySource` is a subclass of `owl:Thing`.

3. *Classes related to environmental and fuel properties*

**`EnvironmentalAndFuelProperties` class**

Description

This class contains individuals and subclasses that are related to fuel types that are used in the district and their related properties, as shown in Figure 43. To keep the figure simple, it does not show any disjoint relationships, but all subclasses of this class are mutually disjoint with each other. As the name suggests, the class also has environmental properties such as specific emissions of fuel and emissions due to biomass fuel transport.

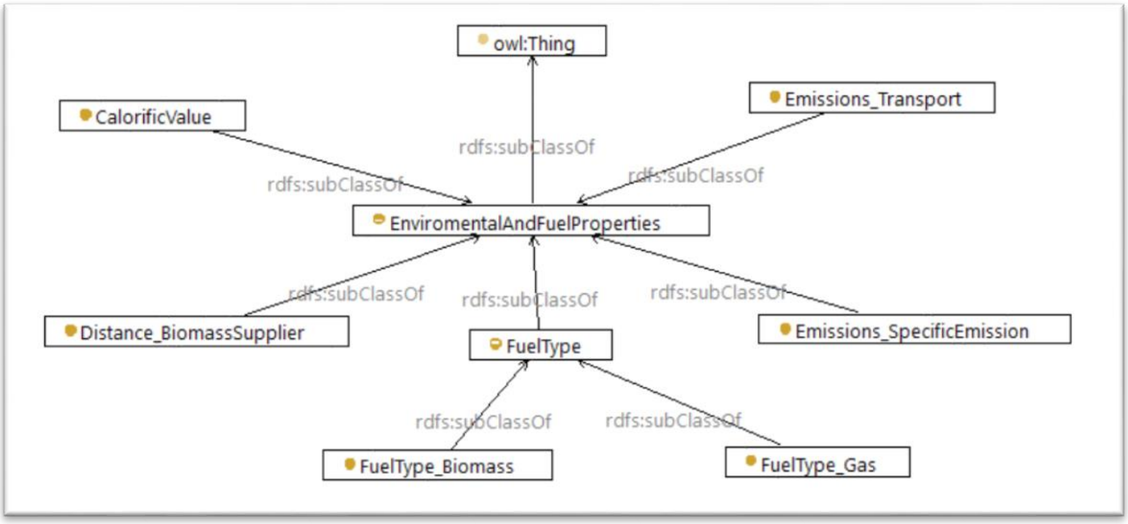



**Figure 43. Subclasses of `EnvironmentalAndFuelProperties` class**

Relations

- This class is a subclass of `owl:Thing`.
- It is equivalent to `socio_technical_systems:EnvironmentalProperty` class

*4. Classes related to energy source design parameters*

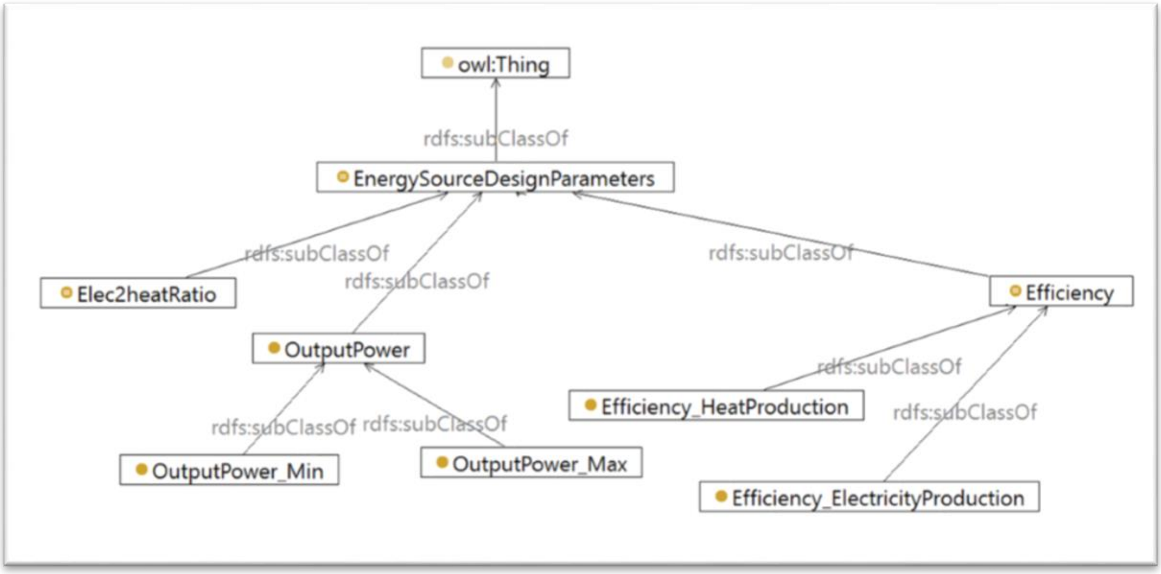



**Figure 44. `EnergySourceDesignParameters` class and its subclasses**

**`EnergySourceDesignParameters` class**

Description

This class contains individuals and subclasses that are related to the various properties of the energy sources in the district including their efficiencies, maximum and minimum power capacities, and electricity to heat power ratios. For clarity, Figure 44 above does not show any disjoint relationships, but all subclasses of this class are mutually disjoint with each other.

Relations

- This class is a subclass of `owl:Thing`.
- It is equivalent to `socio_technical_systems:DesignProperty` class.

*5. Classes related to building operational parameters*

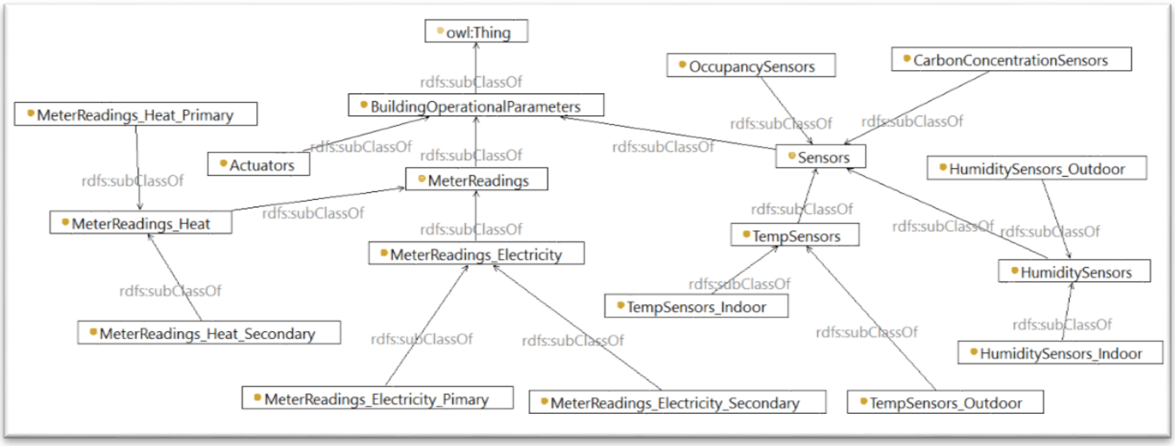



**Figure 45. Classes related to `BuildingOperationalParameters`**

**`BuildingOperationalParameters` class**

<u>Description</u>

This class contains individuals and subclasses that are related to the various operational parameters relevant to real-time operations in the buildings. Various subclasses representing sensors and meter readings are present under this class. For clarity, Figure 45 above does not show any disjoint relationships, but all subclasses of this class are mutually disjoint with each other.

<u>Relations</u>

- This class is a subclass of `owl:Thing`.

*6. Classes related to location of parameters in BMS or EMS*

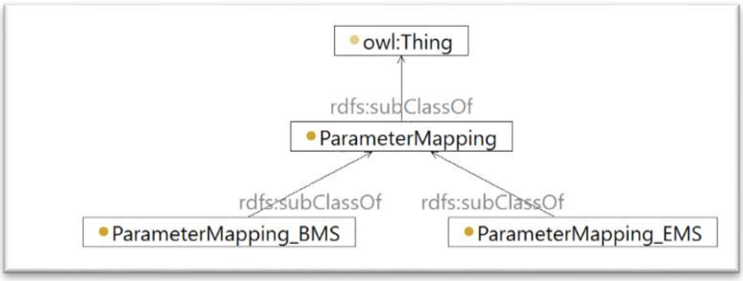



**Figure 46. `ParameterMapping` class and subclasses**

**`ParameterMapping` class**

Description

Real-time energy management requires dynamic values to be read for parameters from a BMS or EMS. This class contains all individuals that represent the BMS/EMS endpoint information, i.e. the location of the real-time values of these parameters. More details can be found when the properties of this class are explained.

Relations

- `ParameterMapping` is a subclass of `owl:Thing`.

7. *Classes related to district operational parameters.*

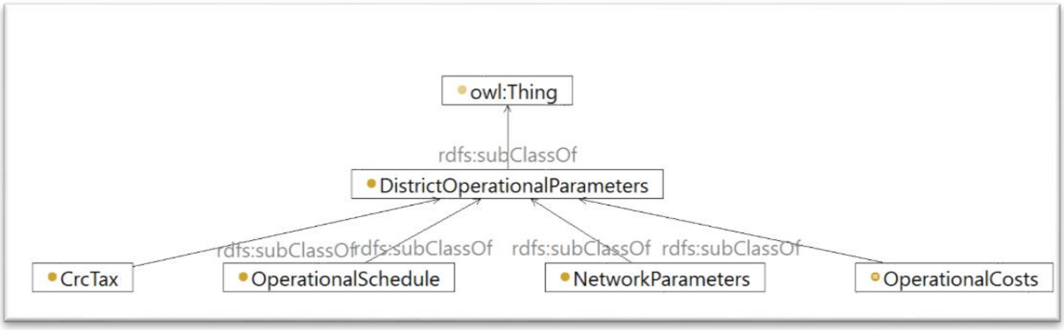



**Figure 47. `DistrictOperationalParameters` class and its subclasses**

**`DistrictOperationalParameters` class**

Description

This class contains individuals and subclasses that are related to operational parameters in the district. Operational parameters in the district are required for energy optimisation at a district level. Figure 47 above shows some of the subclasses of this class; however, it does not show any disjoint relationships between them. All subclasses of this class are mutually disjoint with each other.

Relations

- This class is a subclass of `owl:Thing`.

8. *Classes related to use cases and scenarios for real-time energy management*

**`UseCases` class**

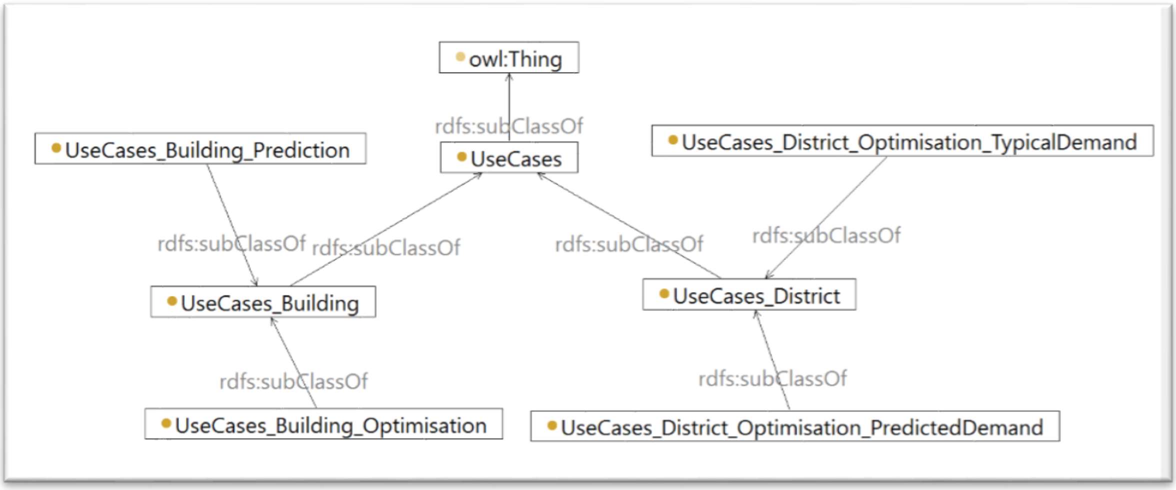



**Figure 48. `UseCases` class and its subclasses**

Description

One of the most important classes of the **REMO** ontology that is needed for real-time energy management applications is the `UseCases` class. This class contains individuals and subclasses that represent the use cases for optimisation and prediction. Once individuals are defined in this class, various parameters (or instances) needed for the prediction and optimisation models are automatically inferred through SPIN rules during the reasoning process (shown in the reasoning Section, 7.1.2). Figure 48 above shows some of the subclasses of this class but does not demonstrate any disjoint relationships between them. All subclasses of this class are mutually disjoint with each other.

Relations

- This class is a subclass of `owl:Thing`.

**`UseCases_Building` class**

Description

This class contains individuals that represent the building-related use cases.

Relations

- This class is a subclass of `UseCases` class.
- Disjoint with `UseCases_District` class.

### `UseCases_Building_Optimisation` class

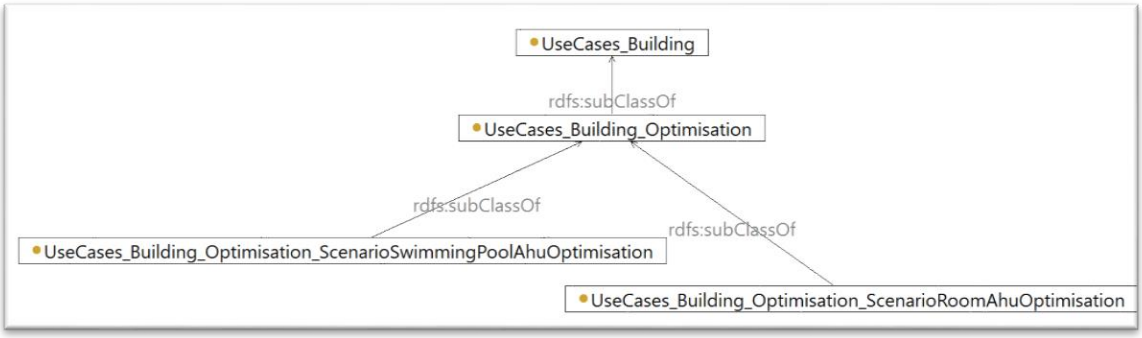



**Figure 49. `UseCases_Building_Optimisation` class and its subclasses**

<u>Description</u>

This class contains the optimisation-related use cases applied within a building. Subclasses of this class are shown above in Figure 49. These are the use cases adopted from the work conducted in the SportE2 project.

<u>Relations</u>

- Subclass of `UseCases_Building`.
- Disjoint with `UseCases_Building_Prediction` class.

Other subclasses of `UseCases_Building_Optimisation` and its description are shown in Table 13 below. Details of this can be found in the appendices.

**Table 13. Other classes related to `UseCases_Building_Optimisation` class**

| Name of class | Description |
|---|---|
| `UseCases_Building_Optimisation_ScenarioSwimmingPoolAhuOptimisation` | This class contains the optimisation-related use cases applied within a building, especially looking into optimisation of the air-handling unit of a zone or space containing a swimming pool. |
| `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` | This class contains the optimisation-related use cases applied within a building, especially looking into optimisation of the air-handling unit of a zone or space. |

### `UseCases_Building_Prediction` class

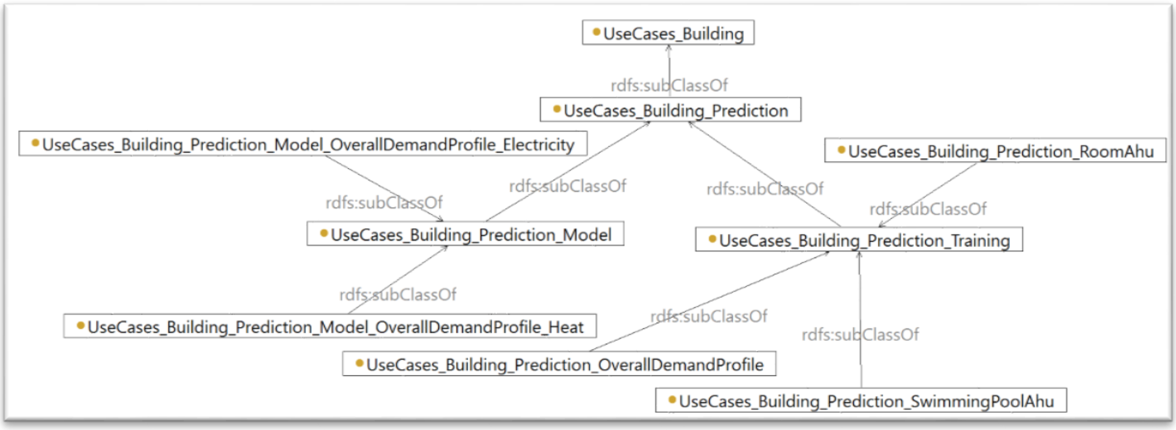



**Figure 50. `UseCases_Building_Prediction` class and its subclasses**

<u>Description</u>

This class contains the prediction-related use cases applied within a building. These use cases can be applied for the training of ANN models for a particular optimisation scenario. Sometimes they are used for running of the actual ANN models. Subclasses are shown in Figure 50 above.

<u>Relations</u>

- Subclass of `UseCases_Building`.
- Disjoint with `UseCases_Building_Optimisation`.

### `UseCases_Building_Prediction_Model` class

<u>Description</u>

This class contains individuals that represent use cases to run the actual ANN model. All the information needed to run the ANN model is provided through the properties of the individuals defined through this class.

<u>Relations</u>

- Subclass of `UseCases_Building_Prediction`.
- Disjoint with `UseCases_Building_Prediction_Training` class.

Other subclasses of `UseCases_Building_Prediction_Model` and its description are shown in Table 14 below. Details of this can be found in the appendices.

**Table 14. Other classes related to `UseCases_Building_Prediction_Model` class**

| Name of class | Description |
|---|---|
| `UseCases_Building_Prediction_Model_OverallDemandProfile_Heat` | Individuals of this class represent use cases that support running of ANN models which predict the overall heat demand profile of buildings. |
| `UseCases_Building_Prediction_Model_OverallDemandProfile_Electricity` | Similar to class `UseCases_Building_Prediction_Model_OverallDemandProfile_Heat`, but here the focus is on electricity demand profiles prediction of buildings and not heat. |

### `UseCases_Building_Prediction_Training` class

Description

This class contains individuals that represent use cases to train the actual ANN model. All the information needed to train the ANN model is provided through the properties of the individuals defined through this class.

Relations

- Subclass of `UseCases_Building_Prediction`.
- Disjoint with `UseCases_Building_Prediction_Model` class.

Other subclasses of `UseCases_Building_Prediction_Training` and their descriptions are shown in Table 15 below. Details of this can be found in the appendices.

**Table 15. Other classes related to `UseCases_Building_Prediction_Training` class**

| Name of class | Description |
|---|---|
| `UseCases_Building_Prediction_SwimmingPoolAhu` | This use case class is needed to support the `UseCases_Building_Optimisaiton_ScenarioSwimmingPoolAhuOptimisation` class, as it provides information for the training of the ANN model that is used for optimisation of air-handling units in spaces or zones containing swimming pools. |
| `UseCases_Building_Prediction_RoomAhu` | This class has individuals that represent use cases class needed to support the `UseCases_Building_Optimisaiton_ScenarioRoomAhuOptimisation` class, as it provides information for the training of the ANN model that is to be used for optimisation of air-handling units in rooms. |

| | |
|---|---|
| UseCases_Building_Prediction_OverallDemandProfile | This class contains individuals that represent the prediction-related use cases applied for a building. The individuals and their properties once reasoned will contain information for developing and training an ANN model that will be applied for day-ahead forecasts of overall building heat and electricity demand profiles. |
| UseCases_Building_Prediction_OverallDemandProfile_Heat | This class contains individuals that represents the prediction models of overall heat demand profiles of buildings. The individuals and their properties once reasoned will contain information for running an ANN model that will be applied for day-ahead forecasts of heat demand of the building. |
| UseCases_Building_Prediction_OverallDemandProfile_Electricity | Similar to class UseCases_Building_Prediction_OverallDemandProfile_Heat, but here the focus is on electricity demand profile prediction of buildings and not heat. |

**UseCases_District class**

<u>Description</u>

This class contains individuals that represent the district optimisation-related use cases. The individuals of these use cases are linked to energy producing buildings through properties, and this consequently provides information for running the district analytical model and optimisation model through reasoning. The detailed application of this class and its individuals is shown through the use cases in Section 5.3.

<u>Relations</u>

- This class is a subclass of UseCases class.
- Disjoint with UseCases_Building class.

Other subclasses of UseCases_District and their descriptions are shown in Table 16 below. Details of this can be found in the appendices.

**Table 16. Other classes related to `UseCases_District` class**

| Name of class | Description |
|---|---|
| `UseCases_District_Optimisation_TypicalDemand` | This class contains the district optimisation use case but uses a typical demand profile for each building in the district. |
| `UseCases_District_Optimisation_PredictedDemand` | This class contains the district optimisation use case but uses predicted demand profile for each building in the district. In other words, the district optimisation uses day-ahead demand forecasts for heat and electricity profiles of each building. |

### 9. *Classes related to optimisation*

#### `Optimisation` class

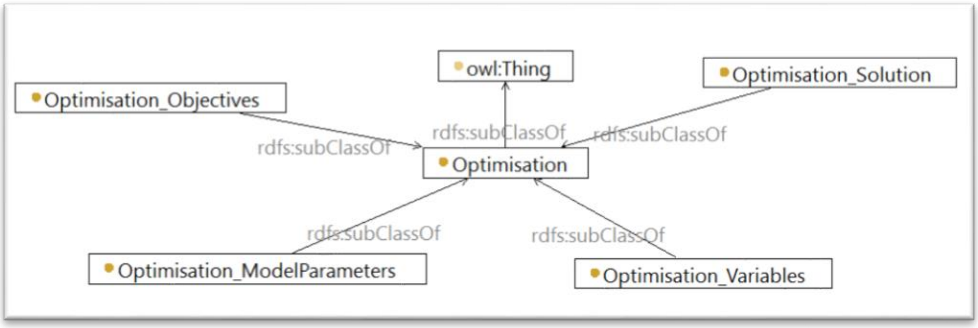



**Figure 51. `Optimisation` class and subclasses**

Description

This class contains individuals and subclasses that are related to concepts needed for real-time energy optimisation. They provide knowledge and information for the optimisation model to run, such as the optimisation variables, the objectives, and the solutions. Optimisation model input and output are mainly presented through the properties associated with the `UseCases` class. Figure 51 above shows the various subclasses of this class. It does not demonstrate any disjoint relationships. All subclasses of this class are mutually disjoint with each other.

Relations

- This class is a subclass of `owl:Thing`.

#### `Optimisation_ModelParameters` class

Description

This class contains individuals that represent the different parameters needed to initiate the district optimisation model. The subclasses of this class are shown in Figure 52 below:

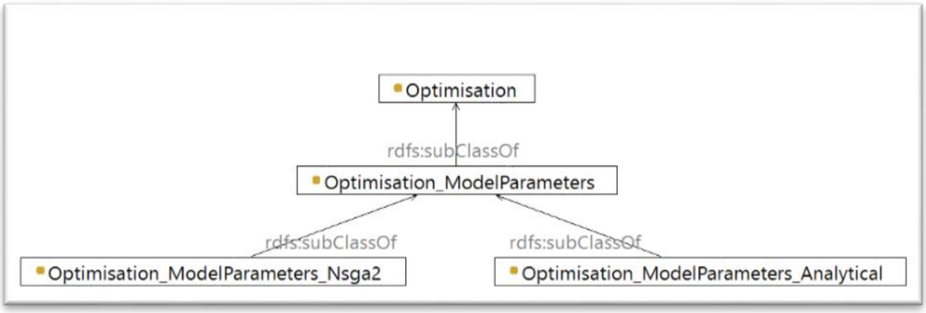



**Figure 52. `Optimisation_ModelParameters` class and its subclasses**

<u>Relations</u>

- This class is a subclass of `Optimisation`.
- Disjoint with sibling classes: `Optimisation_Objectives`, `Optimisation_Solution`, and `Optimisation Variables`.

Other subclasses of `Optimisation_ModelParameters` and their descriptions can be found in the appendices.

### `Optimisation_Objectives` class

<u>Description</u>

This class contains individuals that represent the different objectives of the optimisation problem at both a building and a district level. The subclasses in Figure 53 below show the different types of objectives.

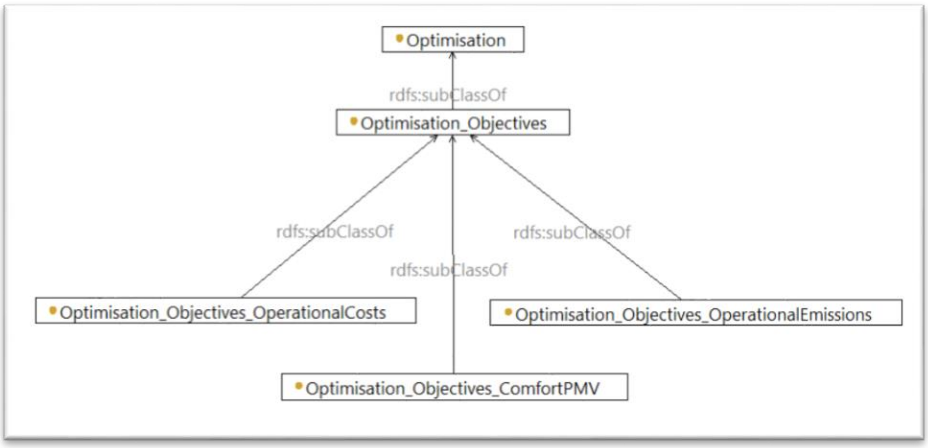



**Figure 53. `Optimisation_Objectives` class and its subclasses**

Relations

- This class is a subclass of `Optimisation`.
- Disjoint with sibling classes: `Optimisation_ModelParameters`, `Optimisation_Solution`, and `Optimisation Variables`.

Other subclasses of `Optimisation_Objectives` and their descriptions can be found in the appendices.

**`Optimisation_Variables` class**

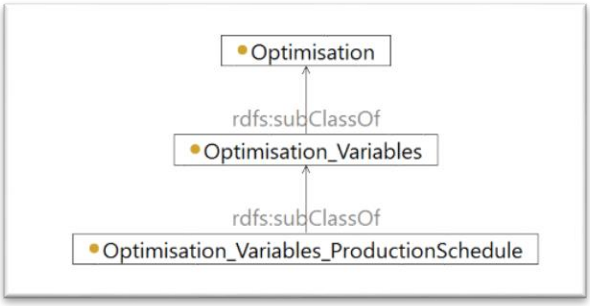



**Figure 54. `Optimisation_Variables` class and its subclasses**

Description

This class contains individuals that represent the optimisation variable or, in other words, the decision variables of the optimisation problem at the district level. The individuals here are automatically constructed when the district energy optimisation use case is defined in the `UseCases_District` class.

Relations

- This class is a subclass of `Optimisation`.
- Disjoint with sibling classes: `Optimisation_ModelParameters`, `Optimisation_Solution`, and `Optimisation_Objectives`.

### 5.2.2. REMO Ontology Properties

This section describes the data properties and object properties in **`REMO`** ontology. Only the most important properties are described in this section; the others can be found in the appendices.

1. Properties Related to `Building` class

**Table 17. Properties related to `Building` class**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasOutdoorTempSensor*** | `Building` | `TempSensors_Outdoor.` |
| ***hasOutdoorHumSensor*** | `Building` | `HumiditySensors_Outdoor.` |
| ***hasMainElecMeter*** | `Building` | `MeterReadings_Electricity_Primary.` |
| ***hasMainHeatMeter*** | `Building` | `MeterReadings_Heat_Primary.` |

Description

***hasOutdoorTempSensor*** – this property assigns an outdoor temperature sensor to individuals of the `Building` class.

***hasOutdoorHumSensor*** – this property assigns an outdoor humidity sensor to individuals of the `Building` class.

***hasMainElecMeter*** – this property assigns a primary electricity meter to individuals of the `Building` class.

***hasMainHeatMeter*** – this property assigns a primary heat meter to individuals of the `Building` class.

2. Properties related to `EnergyConsumerZone` and `EnergyConsumerRoom` class

Description

***hasSensors*** – this property assigns the sensors to energy-consuming rooms within a building.

***hasMeter*** – this property assigns meters to energy-consuming rooms within a building.

***hasComfortPMV*** – this property assigns a comfort PMV parameter to energy-consuming rooms in the building.

***hasEnergyUsingSystem*** – this property assigns energy-using systems such as radiators or air-handling units to energy-consuming rooms in a building. The range here belongs to an imported class of the **`ee-district`** ontology– `eedistrict:EnergyUsingSystem`.

***hasActuators*** - this property assigns actuators to energy using systems.

***hasSubzone*** – this property assigns subzones to energy consumer zones. An energy consumer zone here can be a building or a room within a building as a building can have many zones. Room, however, is the smallest zone and cannot have a subzone. This restriction can be explicitly mentioned in the properties by the following statement:

"*hasSubzone* **exactly** 0".

3. Properties related to `ParameterMapping` class

**Table 18. Properties related to `ParameterMapping` class**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasLocationString_Read*** | `ParameterMapping` | `xsd:string` |
| ***hasLocationString_Write*** | `ParameterMapping` | `xsd:string` |
| ***hasHistoricalDataLocationString*** | `ParameterMapping` | `xsd:string` |

<u>Description</u>

***hasLocationString_Read*** – this property assigns the individuals of the `ParameterMapping` class and its subclasses to its endpoint location in the BMS or EMS. The individuals represent each dynamic parameter of the **`REMO`** ontology whose value might have to be read in real time from the BMS or EMS, for example, temperature sensors, actuator setpoints and so forth. The endpoint location is stored as a string.

***hasLocationString_Write*** – this property is very similar to the property above. However, here the string value assigned represents the endpoint location to modify or write setpoints in the BMS or EMS. For example, the actuators in the building might need to be modified after the optimisation process. Retrieving this information from the ontology, therefore, enables editing of the actuator setpoints in the BMS or EMS.

***hasHistoricalDataLocationString*** – this property assigns the endpoint location of parameters in the BMS or EMS to individuals of the `ParameterMapping` class and its subclasses. Through this string location, the historical data of the parameter of interest can be retrieved.

4. Properties related to `UseCases` class and its subclasses.

**Table 19. Properties related to UseCases class and its subclasses**

| Name of Property | Domain | Range |
|---|---|---|
| ***isApplicableFor*** | `UseCases_Building` | `EnergyConsumerRoom` |
| ***hasOptimObjective*** | `UseCases_District` | `MeterReadings` |
| | | `Optimisation_Objectives` |
| ***hasOptimSettings*** | `UseCases_District` | `Optimisation_ModelParameters_Nsga2` |

| | | |
|---|---|---|
| ***hasOptimModelParameters*** | `UseCases_District` | `Optimisation_ModelParameters_Analytical` |
| ***isApplicableForDistrictOptimisation*** | `UseCases_District` | `EnergyProducerBuilding` |

<u>Description</u>

***isApplicableFor*** – this property assigns the energy management use case (prediction or optimisation) to a particular space or zone in an energy consumer building. This property is at a building level only and is available only to building-level use cases.

***hasOptimObjective*** – this property assigns the objectives of optimisation to the use case. The objectives assigned are automatically inferred through SPIN rules defined in the ontology. Further explanation of this can be found in Section 6.1.2. The objective is usually individuals from the `MeterReadings` class or `Optimisation_Objectives` class.

***hasOptimSettings***– this property assigns the settings for the district optimisation model (for the NSGA-II algorithm) to the individuals of the `UseCases_District` class. The instances defined in the `Optimisation_ModelParameters_Nsga2` class already have default values if the user wants to leave this unchanged.

***hasOptimModelParameters*** – this property infers some of the parameters needed for the district energy optimisation and analytical model to run. The SPIN rules defined in the class `UseCases_District` makes this possible.

The `Optimisation_ModelParameters_Analytical` class already predefines the individuals needed, but the properties of these individuals are needed for the models to run, which are only complete once the ontology is reasoned. Examples of this are shown in Section 6.1.2.

***isApplicableForDistrictOptimisation*** – this property assigns the district energy management use case (optimisation) to a particular energy producer building at the district level. This property can be used only at a district level and is available only to energy producer buildings, which supply energy to the buildings at a district level.

5. Properties related specifically to building prediction and optimisation use cases class

Description

***hasAnnInput*** – this property assigns the inputs of the ANN model to the prediction use case (through subclasses). This property is at a building level only and is available only to building-level use cases. The individuals are automatically assigned to this property through inference, as SPIN rules are embedded in the ontology. These individuals are selected from the range of this property, i.e. from the `Actuators` class, `Sensors` class, or at times even the `ActivitySchedule` Class.

***hasAnnOutput***– this property is similar to property above, but here it assigns the output parameters of the ANN model to the use case. These properties mainly infer the recorded meter readings or other objectives, such as comfort factor.

***hasOptimInput***– this property assigns the input parameters needed for the building energy optimisation model of the selected use case. The individuals allocated to the use case through this property are automatically inferred once the ontology is reasoned.

***hasOptimObjective*** – defined previously under properties related to the `UseCases` class and its subclasses (refer to Table 19).

***hasDecisionVariable*** – this property assigns the decision variables of the optimisation model to the use cases This is applicable for both the building and district optimisation problems. These are again automatically inferred based on knowledge embedded in the ontology through SPIN rules. They are mainly from the class `Actuators` in the case of building optimisation and the class `ProductionScheduleHeat` in the case of district optimisation.

**Table 20. Properties related to `UseCases_Building_Prediction` and `UseCases_Building_Optimisation` class**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasAnnInput*** | `UseCases_Building_Prediction` | `Actuators` |
| | | `ActivitySchedule` |
| | | `Sensors` |
| ***hasAnnOutput*** | `UseCases_Building_Prediction` | `MeterReadings` |
| | | `Optimisation_Objectives` |
| ***hasOptimInput*** | `UseCases_Building_Optimisation` | `Sensors` |
| | | `ActivitySchedule` |
| | | `Actuators` |
| | | `Optimisation_Objectives` |

| | | |
|---|---|---|
| ***hasOptimObjective*** | UseCases_Building_Optimisation | MeterReadings |
| ***hasDecisionVariable*** | UseCases_Building_Optimisation | Actuators |
| | | ProductionScheduleHeat |

6. Properties related to Optimisation class

**Table 21. Properties related to Optimisation class**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasAnalyticalModelValue*** | Optimisation_ModelParameters_Analytical | xsd:integer |
| | | xsd:float |

Description

***hasAnalyticalModelValue*** – this property assigns a numerical value to individuals in the Optimisation_ModelParameters_Analytical class, which is needed for running the district analytical or optimisation model. If not allocated by the user, they are usually inferred through SPIN rules (as shown in Section 6.1.2).

### 5.2.3. Dependencies with the ee-district ontology

**ee-district** ontology is imported into **REMO** ontology as shown earlier in Figure 38. Some of the benefits of doing this were:

- Reusing some of the classes such as UnitOfMeasure (imported into **ee-district** from **system** ontology) and EnergyUsingSystem.
- All the applications that **ee-district** ontology supports have the possibility also to be linked to the overarching framework in this research, if needed in the future.
- **ee-district** allows a detailed description of a district including social and technical entities.
- Concepts from CIM standards that are relevant for energy management systems to talk to each other are included in the **ee-district** ontology, and therefore they are available for **REMO** ontology as well. Using these standards can be beneficial for further development of the framework in the future, to establish communications with the energy management systems (BMS/EMS).

Some of the concepts in **REMO** ontology are similar to **ee-district** ontology concepts. These were mapped to each other using the equivalent property. Some of the links are shown below in Figure 55.

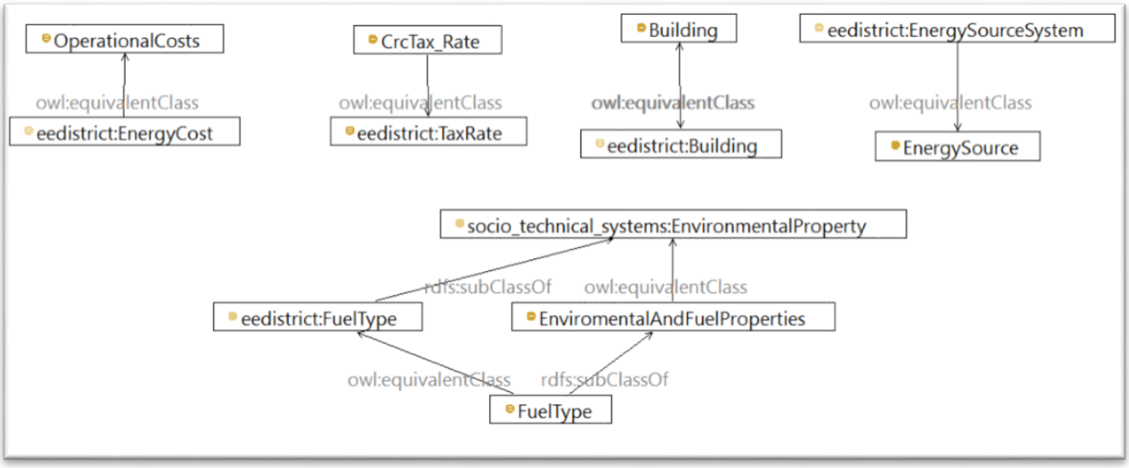



**Figure 55. Links between REMO and ee-district ontology**

`ScalarValue` class and concepts from **ee-district** ontology were also important to adopt into **REMO** ontology to represent the scalar values and properties for various parameters related to real-time energy management. The `ScalarValue` class was imported into **ee-district** through the **system** ontology. Each individual of this class has properties *UnitOfMeaure* and *numericalValue*, which were both linked to **REMO** ontology as shown below in Figure 56 and Table 22.

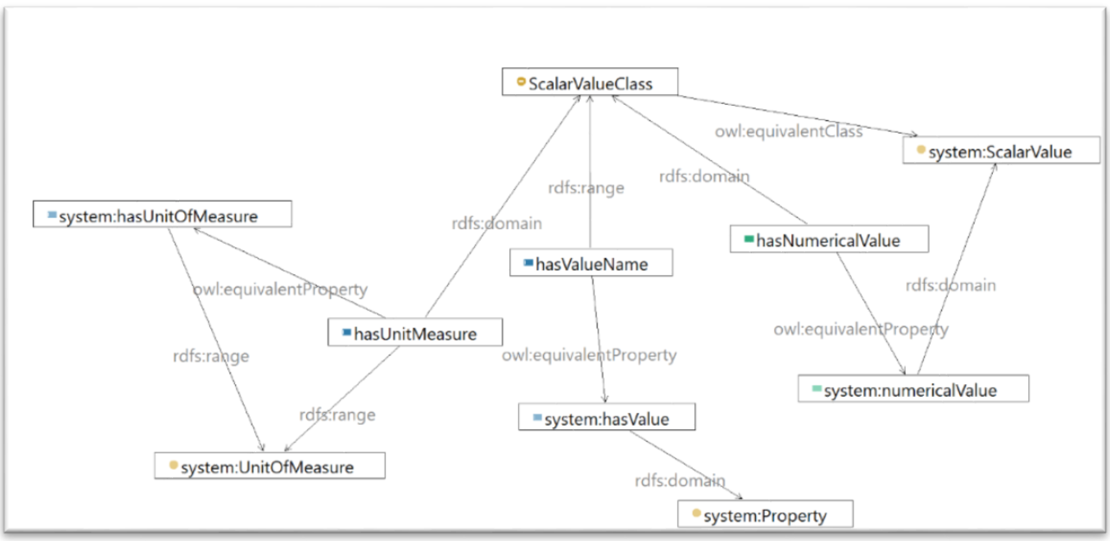



**Figure 56. ScalarValueClass and its properties**

The equivalent property of this class and its purpose is shown below in Table 22.

**Table 22. Equivalent property between `system` and `REMO` ontologies**

| `System` ontology property | `REMO` ontology property | Purpose |
|---|---|---|
| system:hasValue | remo:hasValueName | Represents the scalar value name for each property. |
| system:numericalValue | remo:hasNumerialValue | Stores the numerical value for the property. |
| system:UnitOfMeasure | remo:hasUnitMeasure | Assigns the dimension to the scalar value from the class system:UnitOfMeasure. |

Some classes of **`REMO`** ontology were defined as subclasses of **`ee-district`** ontology. For example, the `EnergyUsingSystem` class belongs to **`ee-district`** ontology, whereas its subclasses belong to **`REMO`** ontology, as shown below in Figure 57. The subclasses are energy-using systems within a building and these concepts were missing in **`ee-district`** ontology.

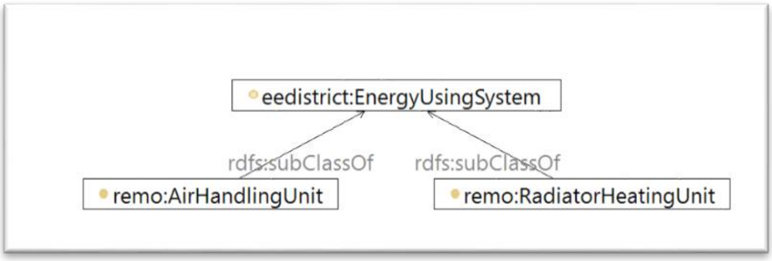



**Figure 57. `eedistrict:EnergyUsingSystem` class and its subclasses in `REMO`**

### 5.2.4. Dependencies with the IFC ontology

*Framework compliance with BIM*

BIM is a methodology increasingly being adopted by the Architecture Engineering and Construction (AEC) industry around the world and especially in the UK. IFC building standards are used as a file format for exchanging information in BIM. Therefore, linking the framework with IFC by importing the **`ifcOWL`** ontology into **`REMO`** ontology was important, as this allows the framework to be applied to all future BIM-compliant construction projects. Integrating the operational stages of the building to BIM is now critical as there are more and more operational data requirements for holistic and real-time energy management. Integrating into BIM process will formalise or standardise the operational data requirement from the conception of the project itself. For example, a COBIE[15] schema for real-time operational energy management can be built from this.

*ifcOWL Ontology – Background*

Previous research has suggested the need to make IFC available as an OWL ontology (Schevers and Drogemuller 2005) so that it can be used with semantic web technologies. Since 2012 there have been many different ifcOWL structures, and there was a need for formalisation and standardisation. Therefore, ifcOWL was created by Pieter Pauwels and Walter Terkaj in close collaboration with buildingSmart and WC3 standardisation bodies (Pauwels and Terkaj 2016). It was developed to support the conversion of IFC files into equivalent RDF files. Currently, only the following formats are supported: IFC4_ADD1, IFC4, IFC2X3_TC1, or IFC2X3_Final schema. However, the **ifcOWL** ontology is in the process of being standardised.

*Using* **ifcOWL** *ontology*

Initial studies on the **ifcOWL** structure showed that some of the concepts were similar to the ones in **REMO** ontology, as shown below in Table 23.

**Table 23. IFC classes and REMO classes**

| **ifcOWL classes** | **REMO classes** |
|---|---|
| IfcBuilding | Building |
| IfcSensor | Sensors |
| IfcActuator | Actuators |
| IfcSpatialZone / IfcSpace | EnergyConsumerRoom |
| IfcZone | EnergyConsumerZone |
| IfcBoiler | GasBoiler / BiomassBoilers |
| IfcDistributionElement | AirHandlingUnit |
| IfcSystem / IfcBuildingSystem | eedistrict:EnergyUsingSystem |

The link can be achieved by explicitly stating the **ifcOWL** classes, and **REMO** classes from table above are equivalent to each other in the domain model. Another possibility is to define rule axioms in the domain model that state that any instance of **ifcOWL**

[15] http://www.bimtaskgroup.org/cobie-uk-2012/

ontology class X is the same as the instance of **REMO** ontology class Y. For example, X here can be the `IfcSensor` class and Y here can be the `Sensors` class.

Many of the concepts from **REMO** ontology are still lacking in the IFC standards today, especially those at a district level. IFC standards, therefore, are not quite good enough to be applied for the real-time energy management approach mentioned in this research, which is also supported by the literature review from Chapter 2. However, some of the concepts included in **REMO** ontology can also be considered to be incorporated into the IFC standards.

The advantage of doing this is that any building in the future that is available in IFC format can be automatically instantiated in **REMO** ontology. This is achieved through the following three steps:

- Building IFC file (office.ifc) is converted to RDF file (officeIFC.rdf) using the tool[16].
- The newly created building IFC owl file (officeIFC.rdf) is imported into instantiated **REMO** ontology (remoInstance.rdf).
- The further reasoning of the **remoInstance** ontology would infer instances from **officeIFC** ontology as instances of the corresponding **remoInstance** ontology classes as they are linked.

Linking with ifcOWL is important as it helps the user in the instantiation process of **REMO** ontology and it does not have to be from scratch. A district can contain many buildings. Thus, in the future, the process mentioned above needs to be automated to bring instances of each building IFC ontology file into instantiated **REMO** ontology. Moreover, the user interface layer could also use the IFC ontology instances to display 3D IFC models of each building on the site.

### 5.2.5. Other dependencies and links

As a part of the action research mentioned earlier in Section 4.1, the author worked on a **SportE2** ontology, which is largely an ontology for buildings that can be used for real-time energy optimisation. Figure 58 shows some of the main concepts identified by the author.

[16] https://github.com/mmlab/IFC-to-RDF-converter/wiki/IFC-to-RDF

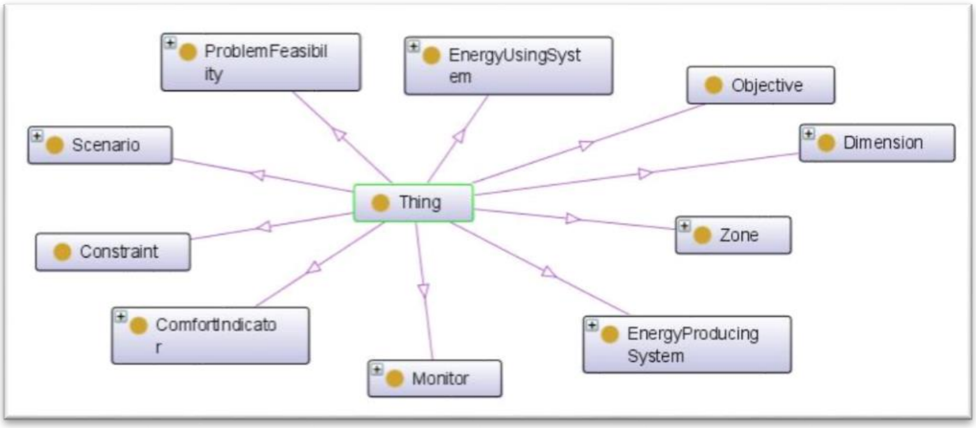



**Figure 58. High-level classes of `SportE2` ontology (Jayan et al. 2014)**

The descriptions of these classes and concepts are mentioned below in Table 24.

**Table 24. `SportE2` ontology classes and their descriptions**

| **Class** | **Description of Class** | **Similar class in `REMO`** |
|---|---|---|
| `Dimension` | Physical quantity measurements. | `system:UnitOfMeasure` |
| `EnergyUsingSystem` | Any system that consumes energy. | `eedistrict:EnergyUsing System` |
| `EnergyProducingSystem` | Any system that produces energy (can also be main grid if no production locally). | `EnergySource` |
| `Scenario` | Covers all the use cases or scenarios to which SportE2 solutions can be applied. | `UseCases` |
| `Zone` | The different zones or rooms or spaces in the sports facilities. | `EnergyConsumerZone` |
| `Objective` | Objectives of scenario. | `Objectives` |
| `Constraints` | All constraints linked with scenario. | - |
| `Monitor` | Sensors and meters involved in a zone to help monitoring. | `Sensors, Meters, Actuators` |
| `ProblemFeasibility` | The individuals here check the feasibility of the optimisation problems and solutions. | - |
| `ComfortIndicator` | Indicates thermal comfort of occupants. | `ComfortPMV` |

Most of the classes described above were adopted into **REMO** ontology, as shown in the table. **SportE2** ontology was not imported into **REMO** because it was not a complete ontology and there were missing concepts and rules, as mentioned earlier in Section 4.1.3.

## 5.3. Generic Use Case Descriptions for the framework

**Use case 1 – Initiate ANN training for use cases that need building energy optimisation**

Description

This is one of the primary use cases, and it is required for some of the other use cases to work (such as use cases 2 and 5). It is applied for the training of ANN models that will consequently be used by the building optimisation use cases.

Pre-requisites

- Existing historical data of buildings, to which these ANN models need to be applied, should be stored in the BMS. If historical data is not available, a simulation model should be able to provide the data for initial training of the models.

Notes

- The timestamp information for historical data is also necessary for the training, but this can be retrieved from the data recordings as most BMS systems will by default have timestamp information associated with them. Post-processing algorithms can be then used to be able to extract day type, month, and hour from this timestamp information. Day type represents all the days of the week as numbers from 1-7.

**Use case 2 – Optimisation of building-related use cases**

Description

This use case is applied to rooms or zones in buildings that need real-time energy optimisation. This use case has been adopted from the optimisation methodology adopted in the SportE2 project (as explained in Section 4.1). Here, the ANN models trained under use case 1 are used as the cost function for the optimisation, and therefore ANN models are a pre-requisite. Since the optimisation is applied in real time, ICT devices that read real-time information for the optimisation input parameters are needed. In reality, this use case can be triggered every 15 minutes or 30 minutes to read data from BMS and consequently optimise the decision variables of the problem. The control of the decision variables of the problem should be possible through the BMS.

Pre-requisites

- Use case 1 needs to be executed prior to this, and consequent ANN models need to be ready for use. (Note: If historical data is not available and Use case 1 cannot be deployed in reality, then a simulation model for each building can be used to develop ANN models.)
- ICT devices for reading real-time data are needed for providing real-time data for optimisation input parameters.
- Control of actuators (optimisation decision variables) should be feasible.

Notes

- The timestamp information is also important here, but this can be processed as ANN input based on the current timestamp reading from the BMS.

**Use case 3 – District production schedule optimisation using typical demand schedules of each building**

Description

This use case is applied at a district supply level and the optimisation applied here looks to produce optimum production schedules for the various energy sources in the district that supply energy to the entire district. The optimisation uses the district analytical model (refer to Section 4.2.2.) as the cost function. This use case is based on the district optimisation work illustrated in Section 4.2.2. This use case uses typical demand profiles of buildings which can be retrieved from the BMS of each building.

Pre-requisites

- District optimisation model needs to be ready for implementation.
- Typical heat and electricity demand profiles are needed for each building that is part of the optimisation problem (i.e. each building part of the district energy model).

Notes

- Currently, this use case only looks into heat production optimisation in the district, because of the focus on district heating, but, in the future, electricity energy production optimisation can also be added in a similar way.
- The schedules here are assumed to be 24-hour with a 30-minute time interval.

**Use case 4 – Initiate ANN training for overall demand prediction of each consumer building in the district**

Description

This use case provides information for training of ANN models for each building, through which day-ahead forecasts of heat and electricity demands can be possible. These day-ahead forecasts can replace the typical demand profiles (of each building) used by the district analytical and optimisation model in use case 3 above. The use case can only be applied if adequate historical data or simulation data is available for training.

Pre-requisites

- The historical data of the buildings, to which optimisation needs to be applied, should be stored in the BMS and must be adequate.

Notes

- The timestamp information is needed, but this is retrieved from the historical data recordings using post-processing algorithms similar to the ones used in use case 1.

**Use case 5 – Running of prediction models to predict overall demand of buildings**

Description

This use case is needed for running the ANN models developed in use case 4. For each building, the day-ahead weather forecasts that are available from the BMS (24-hour profile of outdoor temperature and outdoor humidity) and the building's primary meter readings for the current day (24-hour schedule) are the input variables of the ANN model. Each building will have two ANN models – one for heat demand and one for electricity demand. The ANN models, consequently, provide day-ahead forecasts of demand profiles for the considered building.

Pre-requisites

- Use case 4 needs to be executed before running this use case.
- Heat and electricity meters are needed for monitoring the demand in each building.
- BMS/EMS should be able to forecast weather schedules.

Notes

- The timestamp information is also important here, but this can be processed as ANN input based on the current timestamp reading from the BMS.

**Use case 6 – District optimisation using the predicted demand schedules of each building**

Description

This use case is similar to use case 3; however, the difference is that it uses the predicted demand profiles (from use case 5) of each building rather than using the typical demand profiles. In reality, this use case, once implemented, allows harmonised district energy and building energy optimisation. The facility managers will be able to optimise their production schedules a day ahead, taking into account the demand forecasts of each building (demand forecasts made possible through use case 5). The demand forecasts, on the other hand, consider the daily optimised building energy demand (which is the results of running use case 2).

Pre-requisites

- Use case 4 is needed for the day-ahead demand forecast models to be trained for each building.
- Use case 5 needs to be executed to run the ANN models and compute the forecasted demand for each building.
- District optimisation model needs to be ready for implementation.
- Production schedules of energy sources in the district, once optimised, should be able to be modified in the EMS.

Notes

- Currently, this use case only looks into heat production in the district, because of the focus on district heating in the site considered, but, in the future, use cases for electricity production optimisation can also be added in a similar way.

- The schedule here is assumed to be of 24-hours' duration with a 30-minute time interval.

Use cases 2, 5 and 6 executed one after the other can fulfil the holistic district energy optimisation requirements illustrated earlier in Figure 34 in Section 4.3, combining both building demand side optimisation and supply side optimisation in real time.

The workflow below shows how this is implemented in reality:

1. Use case 1 and use case 4 are executed to train ANN models.
2. Building energy optimisation use cases are applied in real time every 15 or 30 minutes using use case 2.
3. Around midnight the ANN models for predicting overall heat and electricity demand for each building are triggered by using weather forecasts (24-hour profile for outdoor temperature and outdoor humidity) for the next day and each building's previous day's 24-hour demand.
4. District optimisation can then be run as described in use case 6, using the predicted demand profiles from step 3.

The workflow is presented below in Figure 59:

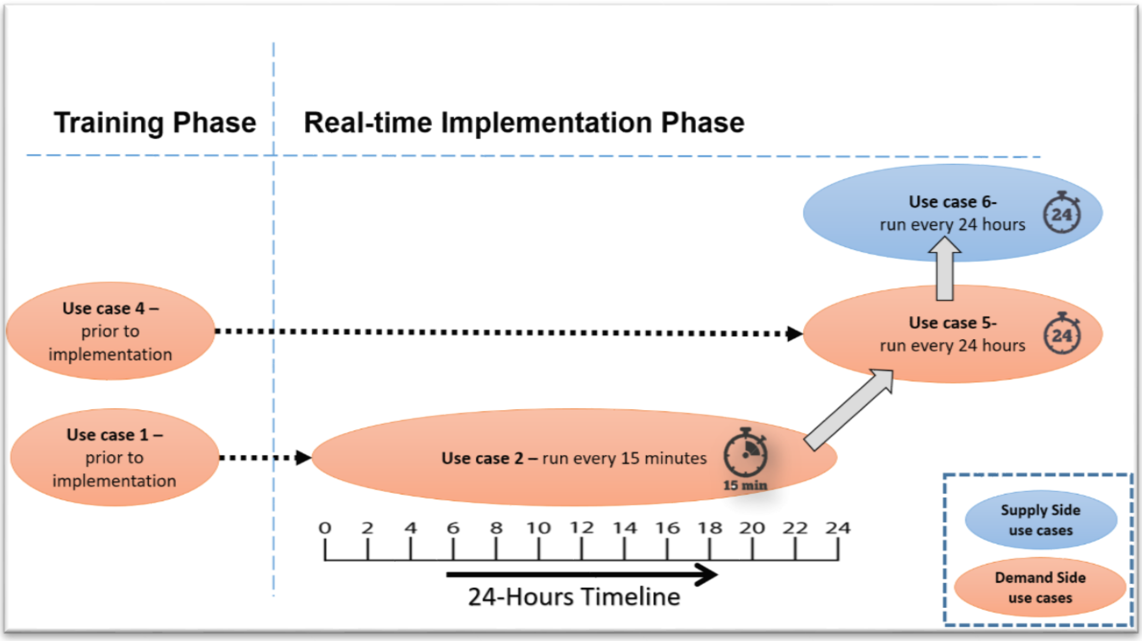



**Figure 59. Workflow of real-time holistic energy optimisation using the use cases**

These use cases are developed from the author's experience through the action research conducted. Looking into other projects in this domain as case studies can help develop more use cases for the framework, by adding more knowledge to **REMO** ontology.

# 6. System Development and Implementation

This chapter focuses on the development and implementation of the overall framework. The focus of this research is the development of the core aspect of the framework, which is the **REMO** ontology. The implementation of the functional layer (demand side optimisation and supply side optimisation) and visual layer is also presented briefly in this chapter. The final part of this chapter describes the running of use cases outlined in Section 5.3.

## 6.1. Implementation of semantic layer (**REMO** ontology)

This section introduces the further development of the ontology by adding rule axioms and constructors. Rule axioms are added to capture the generic knowledge behind the artificial intelligence models in the domain model. These rules in the domain model are important as they enable the reuse of this knowledge in future instance models, i.e. for other similar sites. Constructors, on the other hand, are defined to help the users' instantiation process, making it more efficient. They by default create linked instances and assign them as needed (examples shown in Section 6.1.2). Instantiation of the ontology based on the Resilient project case study is detailed in this chapter in Section 6.1.3. The instantiated ontology is then further used for validation purposes in Chapter 7.

### 6.1.1. Competency questions

Table 25 below lists a few of these questions and they are categorised based on the area of application.

**Table 25. Competency questions**

| **Optimisation-related questions** |
|---|
| What are the various optimisation input parameters given a <use case> |
| What are the various optimisation objectives given a <use case> |
| What are the decision variables given a <use case> |
| What are the optimisation-related settings for district optimisation model given a <use case> |
| What are the analytical model parameters needed for district optimisation given a <use case> |
| **Prediction-related questions** |
| What are the various ANN inputs needed for training the ANN model given a <use case> |
| What are the various ANN outputs needed for training the ANN model given a <use case> |
| What are the various ANN inputs needed for running the ANN model for overall building demand given a <use case> |
| **District-related questions – static topology** |
| List the number of energy consumers in the district |

| |
|---|
| List the number of energy producers in the district |
| List the energy sources in the energy producer building. |
| List the energy sources that supply electricity given an <energy consuming building> |
| List the energy sources that supply heat given an <energy consuming building> |
| **Energy sources and fuel properties-related questions – static information** |
| List the maximum or minimum output power given an <energy source> |
| List the maintenance cost given an <energy source> |
| List the fuel type given an <energy source> |
| List the specific emissions given a <fuel type> |
| **Numerical values and dimensions-related questions** |
| List the numerical value given <scalar value name> |
| List the dimension given <scalar value name> |
| List the scalar value name given <parameter> |
| **Dynamic information and parameter mapping-related questions** |
| List all sensors given <room> |
| List all actuators given <energy using system> |
| List device location parameter name in BMS or EMS given <device> |
| List string location in BMS or EMS to read dynamic real-time information given <device location parameter name> |
| List historical data location in BMS or EMS to retrieve historical data given <device location parameter name> |
| List string location in BMS or EMS to modify setpoints for actuators given <device location parameter name> |

### 6.1.2. Rules and constructors

*SPIN rules embedded in the ontology*

SPIN rules have been used in **REMO** ontology for various purposes, as discussed below:

- To embed the generic knowledge behind optimisation and prediction models used for energy management use cases of the framework. Embedding this knowledge ensures automatic replication of the models, without any human expertise, when **REMO** is applied to future districts and buildings similar in nature.
- Used as constructors which help the instantiation process.
- Used for constraint checking.
- Used to infer numerical values needed for the district analytical model and its optimisation. The user does not explicitly instantiate these numerical values in the ontology, and hence SPIN rules are useful.

The SPIN rules are attached to each class, and they are applied to every individual in that class or its subclasses. Some of the rule axiom definitions in **REMO** and their purpose are explained below.

**Rules relevant for building prediction use cases**

The following SPIN rules are attached to the class `UseCases_Building_Prediction_RoomAhu`. The rules defined under this class infer the input and output (of ANN models) for each individual of the class. The following rules, shown in Table 26, are defined under this class:

**Table 26. Rules used to infer ANN inputs of the room air-handling unit scenario**

| Rule in SPIN language | Algorithm |
|---|---|
| **CONSTRUCT** {<br>?uc :hasAnnInput ?x .<br>}<br>**WHERE** {<br>?uc rdf:type :UseCases_Building_Prediction_RoomAhu .<br>?uc :isApplicableFor ?room .<br>?room :hasEnergyUsingSystem ?system .<br>?system :hasActuators ?x . } | uc *has Ann Input* x<br>IF uc *belongs to* `UseCases_Building_Prediction_RoomAhu` class<br>AND uc *is applicable for* room.<br>AND room *has Energy Using System* system.<br>AND system *has Actuators* x |
| **CONSTRUCT** {<br>?uc :hasAnnInput ?x .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_RoomAhu .<br>?uc :isApplicableFor ?room .<br>?room :hasSensors ?x .<br>} | uc *has Ann Input* x<br>IF uc *belongs to* `UseCases_Building_Prediction_RoomAhu` class<br>AND uc *is applicable for* room.<br>AND room *has Sensors* x. |

Table 26 above infers the ANN input of the prediction model needed for the optimisation of the air-handling unit scenario. ANN input of this particular building-based scenario includes the temperature and humidity sensors in the room, and also the actuators of the energy-using system in the room, as concluded from the SportE2 project knowledge. The input and output variables needed for ANN models are inferred using the various properties of the ontology, as shown in Table 26, which link the room (or zone where the use case is applied), and the sensors and devices in the room. The output parameters of the ANN model are inferred as shown below in Table 27. Here, it is mainly the meters and comfort parameters associated with the targeted room in the building that are inferred.

**Table 27. Rules used to infer ANN outputs of the room air-handling unit scenario**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasAnnOutput ?comfort .<br>?uc :hasAnnOutput ?meter .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_RoomAhu .<br>?uc :isApplicableFor ?room .<br>?room :hasMeter ?meter .<br>?room :hasComfortPMV ?comfort .<br>} | uc *has Ann Output* comfort<br>AND uc *has Ann Output* meter<br><br>IF uc *belongs to* `UseCases_Building_Prediction_RoomAhu` class<br>AND uc *is applicable for* room.<br>AND room *has Meter* meter.<br>AND room *has ComfortPMV* comfort |

Reasoning the ontology therefore assigns the various input and output variables for the individuals of the `UseCases_Building_Prediction_RoomAhu` class. Results of reasoning are shown in Section 7.1.2. SPARQL query can be used to retrieve further information about these input and output variables, as shown in Section 7.1.3 in the validation chapter.

Similarly, rules can be applied to all the other prediction use cases (i.e. subclasses of `UseCases_Building_Prediction`) of **REMO** ontology.

**Rules relevant for building optimisation use cases**

These sets of rules are defined in the class `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation`. Optimisation input is inferred as instances of the `Sensors` and `Actuators` classes, similar to the prediction use case (prediction use case for room air-handling unit ANN model training); the room to which the use case is applied is used as a link to infer the sensor and actuators present here. Table 28 below shows rules used to inference the optimisation inputs needed for the air handling unit optimisation scenario.

**Table 28. Rules used for inferencing the optimisation inputs of the room air-handling unit scenario**

| Rule in SPIN language | Explanation |
|---|---|
| CONSTRUCT {<br>?uc :hasOptimInput ?sensors .<br>}<br>WHERE {<br>?uc a :UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation .<br>?uc :isApplicableFor ?room .<br>?room :hasSensors ?sensors .<br>} | uc *has optim input* sensors.<br><br>IF uc *belongs to* `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class<br>AND uc *is applicable for* Room.<br>AND room *has Sensors* sensors. |
| **CONSTRUCT** {<br>?uc :hasOptimInput ?actuators .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation .<br>?uc :isApplicableFor ?room .<br>?room :hasEnergyUsingSystem ?system .<br>?system :hasActuators ?actuators .<br>} | uc *has optim input* actuators<br>IF uc *belongs to* `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class<br><br>AND uc *is applicable for* room<br><br>AND room *has energy using system* system<br><br>AND system *has Actuators* actuators. |

The decision variables of the optimisation problem, on the other hand, are inferred by checking the energy-using systems in the room and their associated actuators, as shown below in Table 29.

**Table 29. Rules used for inferencing the decision variables of the room air-handling unit scenario**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasDecisionVariable ?dv .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation .<br>?uc :isApplicableFor ?Room .<br>?room :hasEnergyUsingSystem ?system .<br>?system :hasActuators ?dv .<br>} | uc *has Decision Variable* dv<br><br>IF uc *belongs to* `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class<br><br>AND uc *is applicable for* room<br><br>AND room *has energy using system* system<br><br>AND system *has Actuators* dv |

The objectives of the optimisation are the same as the output of the ANN model of the prediction use case. They are inferred by querying which sub-meters and comfort factors are linked to the room, as shown below in Table 30.

**Table 30. Rules used for inferencing the optimisation objectives for the room air-handling unit scenario**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasOptimObjective ?comfort .<br>?uc :hasOptimObjective ?meter .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation .<br>?uc :isApplicableFor ?room .<br>?room :hasMeter ?meter .<br>?room :hasComfortPMV ?comfort .<br>} | uc *has optim objective* comfort<br>AND uc *has optim objective* meter<br><br>IF uc *belongs to* `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class<br><br>AND uc *is applicable for* room<br><br>AND room *has Meter* meter<br><br>AND room *has Comfort PMV* comfort |

Similarly, rules can be attached for other subclasses of the `UseCases_Building_Optimisation` class which can represent other scenarios such as air-handling unit optimisation of swimming pool area. Similarly, rules applied for the other optimisation and prediction use cases are also defined, details of which can be found in appendix A.

**SPIN rules used for inferring some of the numerical values required for the district optimisation model**

The analytical model requires information in order to run, most of which can be obtained by querying the relevant individuals from the ontology through SPARQL queries. A few of these individuals are defined under the class `Optimisation_ModelParameters_Analytical`, and their numerical values need to be inferred using SPIN rules. These rules are defined under the class `Optimisation_ModelParameters_Analytical`.

For example, the number of consumers in the district is represented by one of the individuals under this class and is needed for the analytical model. The numerical value for this can be inferred through the rule shown below in Table 31. The individual "NbOfConsumers" will therefore be assigned a numerical value through the property *hasAnalyticalModelValue* after the reasoning process.

**Table 31. Rules used to infer numerical values needed for district optimisation.**

| Rule in SPIN language | Algorithm |
| --- | --- |
| CONSTRUCT {<br>?individual :hasAnalyticalModelValue ?count .<br>}<br>WHERE {<br>{<br>SELECT ((COUNT(DISTINCT ?consumers)) AS ?count)<br>WHERE {<br>?consumers a :EnergyConsumerBuilding .<br>}<br>} .<br>?individual a :Optimisation_ModelParameters_Analytical .<br>FILTER regex(str(?individual), "NbOfConsumers") .<br>} | Individual *has Analytical model value* count<br><br>WHERE<br>{<br>DISTINCT Number of consumers is count<br>AND consumers *belong to* `EnergyConsumerBuilding` class<br>}<br>AND individual *belongs to* `Optimisation_ModelParameters_Analytical` class<br><br>FILTER individual *with name* "NbOfConsumers" |

*<u>Constructors defined in the ontology</u>*

Constructors are added to make the instantiation process of the ontology easier. These constructors are SPARQL queries that can add initial values to the instance being created. Some of the applications of constructors in **REMO** ontology are shown below:

**Creating scalar value for parameters**

When an individual representing 'distance to biomass supplier' is instantiated under the class `Distance_BiomassSupplier`, constructors are defined to create a scalar value for this individual automatically under the class `ScalarValueClass`. These constructors make it easier for the user to instantiate the ontology, and they consequently would only need to assign a dimension and numerical value to this scalar value instance defined. The constructor to do this would be defined under the `Distance_Biomass` class, as shown below in Table 32.

**Table 32. Constructors defined for `Distance_Biomass` class**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?this :hasValueName ?new .<br>?new a :ScalarValueClass .<br>}<br>**WHERE** {<br>?this a :Distance_BiomassSupplier .<br>**BIND** (str(?this) **AS** ?name) .<br>**BIND** (STRAFTER(?name, "#") **AS** ?stringName) .<br>**BIND** (STRBEFORE(?name, "#") **AS** ?uri) .<br>**BIND** (URI(**CONCAT**(?uri,"#ScalarValue_", ?stringName)) **AS** ?new) .<br>} | this *has value name* new<br><br>AND new *belongs to* `ScalarValueClass`<br><br>IF<br>this *belongs to* `Distance_BiomassSupplier` class<br><br>AND BIND (string value of variable (this)) *AS* name<br><br>AND BIND (string which comes after "#" in name) AS stringName<br><br>AND BIND (string which comes before "#" in name) AS uri<br><br>AND BIND (concatenate strings: uri, "#ScalarValue_", stringName ) AS new |

The constructor above creates an individual under the `ScalarValueClass` for every individual created under the `Distance_BiomassSupplier` class.

The variable 'this'[17] during run-time assigns itself with instances of class and its subclasses.

The 'BIND' command here assigns a value to a variable from the basic graph pattern or property path expression. In the example above, the string value of the instance in 'this' is assigned to the variable 'name'.

The STRAFTER command is used in the form:

STRAFTER (?name, "#").

This command returns a string value of the variable in 'name' following the '#' symbol. In Table 32, the string value of the variable 'name' is the URI of the individual which in this case is "http://www.resilient-project.eu/theworks#TransportEmission_BiomassBoiler_1".

[17] https://www.w3.org/Submission/spin-modeling/#spin-rules-thisUnbound

Therefore, the STRAFTER command applied above would return "TransportEmission_BiomassBoiler_1". The STRBEFORE command is used in the form:

STRBEFORE (?name, "#")

This command returns a string value of the variable in 'name' prior to the '#' symbol. Therefore the STRBEFORE command applied above would return "http://www.resilient-project.eu/theworks".

The CONCAT command is used here to join two or more strings as one. In the example shown in Table 32, string value of variable 'uri', '#ScalarValue_' and the string value of variable 'stringName' are combined. The resultant string stored in the variable 'new' is "http://www.resilient-project.eu/theworks#ScalarValue_Emissions_Transport_BiomassBoiler_1"

Many other constructors similar to this can be applied to **REMO** ontology, which simplifies the instantiation process for the user while creating an instance model. Some further examples can be found under Appendix A.

### 6.1.3. Instantiation

In this research, the **REMO** ontology is applied to the site in Ebbw Vale to create an instance model which is named **theworks** ontology, as shown below in Figure 60.

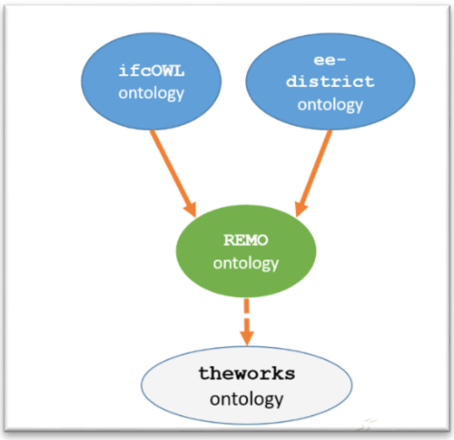



**Figure 60. theworks ontology instantiation**

The ontology is instantiated with all the possible data collected from the site. There are four stages to instantiation, as listed below:

1. **District- and building-related data**

The following elements within the district are instantiated or defined here:

- Buildings in the district as either energy-producing or -consuming buildings.
- All the energy-consuming zones and rooms within each building.
- Energy using systems in the rooms or zones.
- Actuators linked to energy using systems.
- Sensors and meters associated with the rooms or zones.
- Demand schedules for each building in the district.
- Energy sources in the district, which can be centralised sources (National Grid for electricity) or decentralised sources (such as biomass boilers, gas boilers or combined heat and power units).
- The energy source design properties such as maximum and minimum output power, maintenance costs, electricity to heat ratios and so forth. Most of these are defined through constructors. An example is shown below:
    - A new biomass boiler instance named 'BiomassBoiler_1' is instantiated under the `BiomassBoiler` class, as shown below in Figure 61:

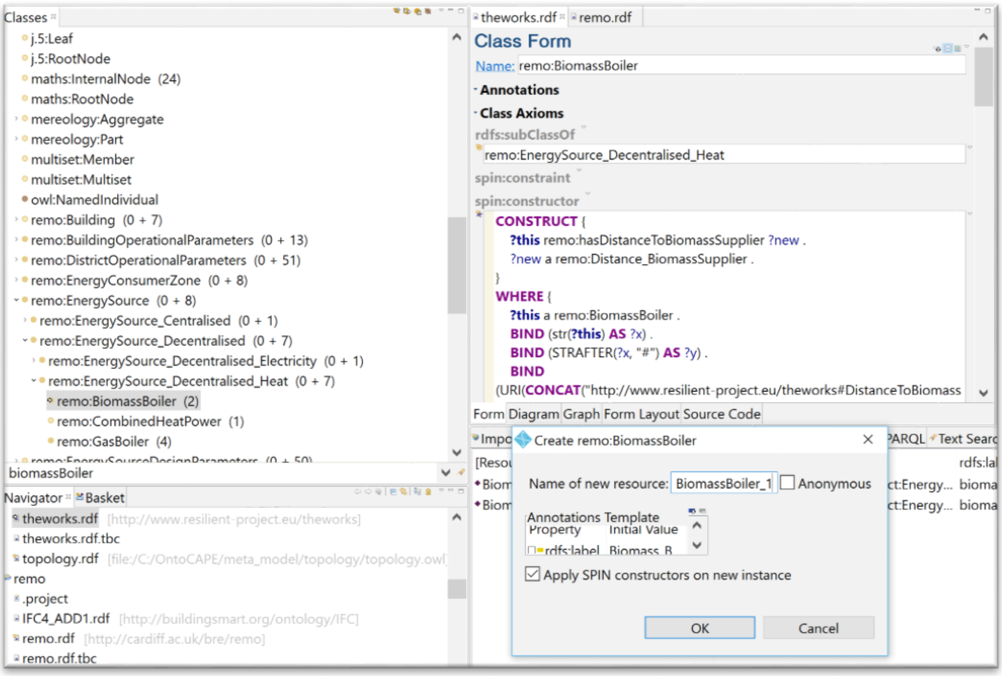


**Figure 61. Instance of `BiomassBoiler` created**

- 
    - The constructor defined under the class `BiomassBoiler` initialises individuals for the various properties of a biomass boiler and associates them to the individual 'BiomassBoiler_1', as shown below in Figure 62:

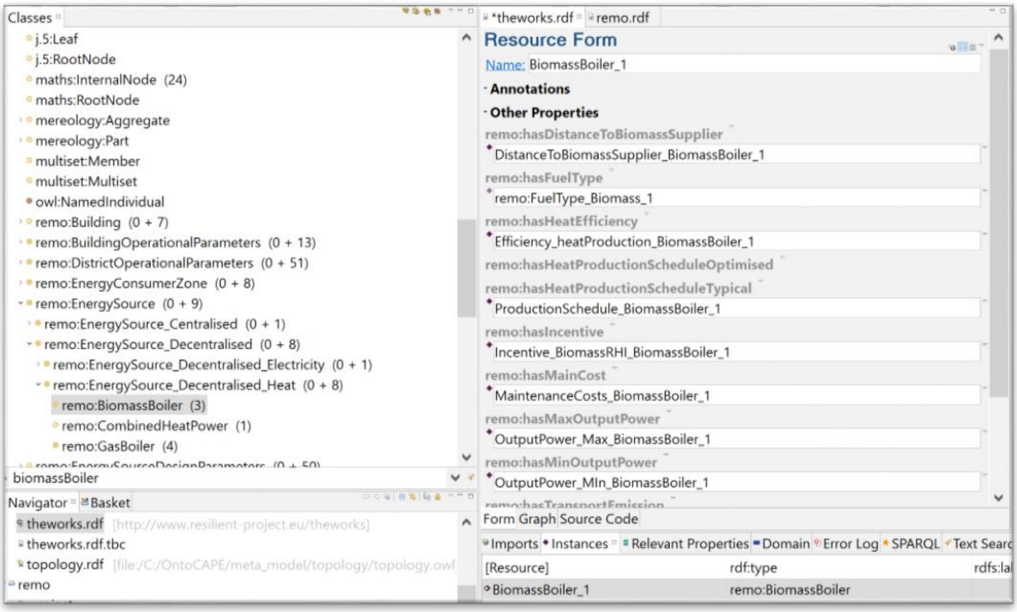



**Figure 62. Instance of `BiomassBoiler` properties initialised and assigned**

- Energy prices and costs in the district.
- Production schedules for energy sources in the district.
- Environmental and fuel properties such as fuel type, specific emissions, fuel transport emissions, calorific value and so forth.
- Optimisation-related parameters have been pre-defined instances, and some have default values. The values of these instances need to be updated by the user as needed.

**2. Parameter mapping data from BMS/EMS**

- Certain parameters in the ontology, especially operational parameters with dynamic values (changes in real-time), require a reference to their endpoint location in BMS or EMS systems. This reference is represented by individuals under the `ParameterMapping` class and is automatically created through constructors when the user defines the parameters (as shown in Section 6.1.2 using constructors). The string values for these references/individuals need to be assigned by the user. An example of this is shown below:
    - A new actuator instance named 'Actuators_LeisureCentre_SupplyAirTemp' under the `Actuators` class is defined as shown below in Figure 63:

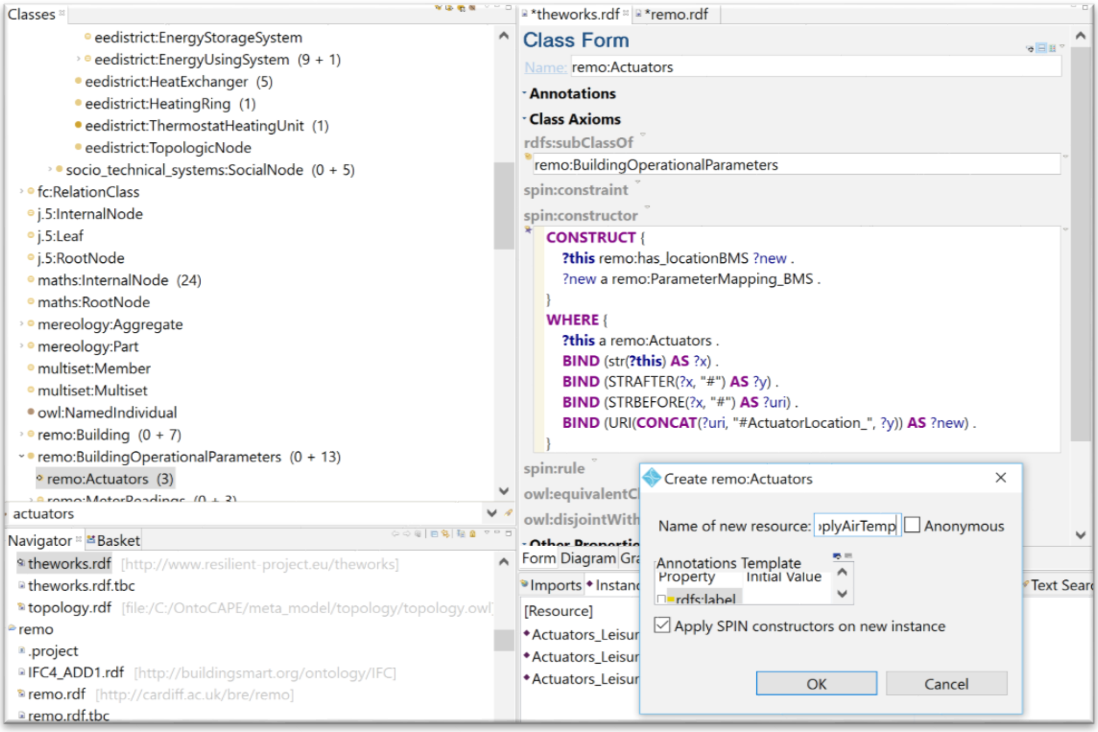


**Figure 63. Actuator instance being created**

- The constructor defined under the class `Actuators` creates an individual that represents the endpoint location of the actuator in BMS and associates this new individual to ‘Actuators_LeisureCentre_SupplyAirTemp’ using the property *has_locationBMS*, as shown below in Figure 64:

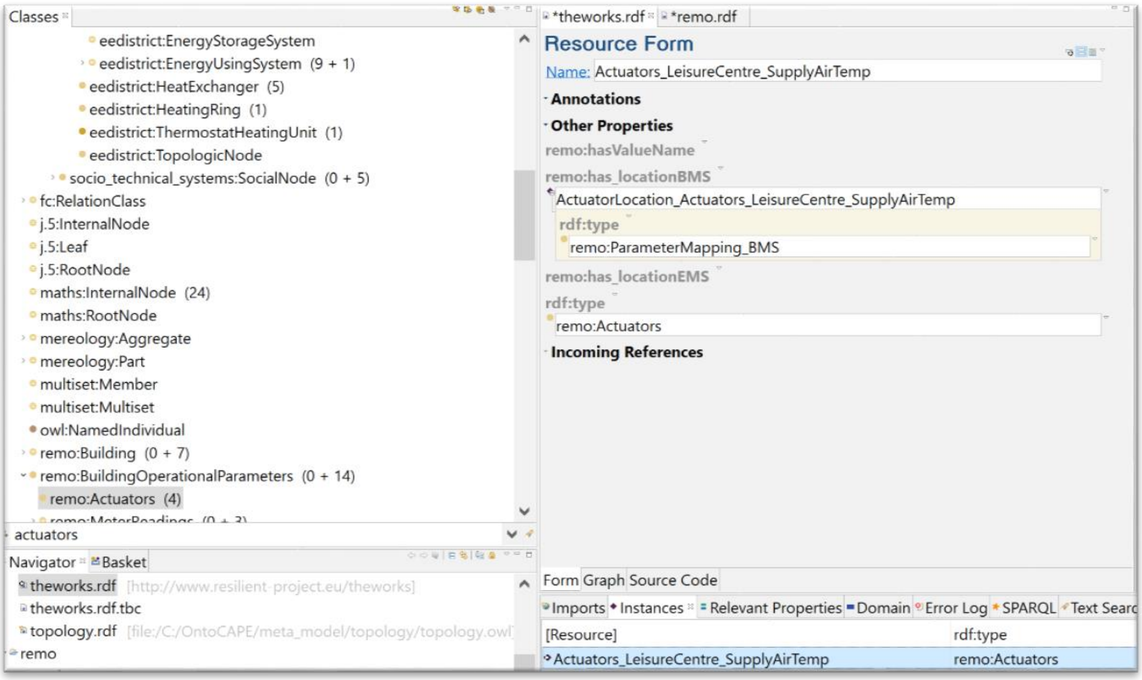


**Figure 64. Actuator instance being assigned an individual that represents the endpoint location**

- The individual is then initialised with a string value that represents its locations in the BMS or EMS, from which relevant information can be retrieved. In the case of actuators, the historical data, real-time data, and setpoint location can all be accessed, as shown below in Figure 65:

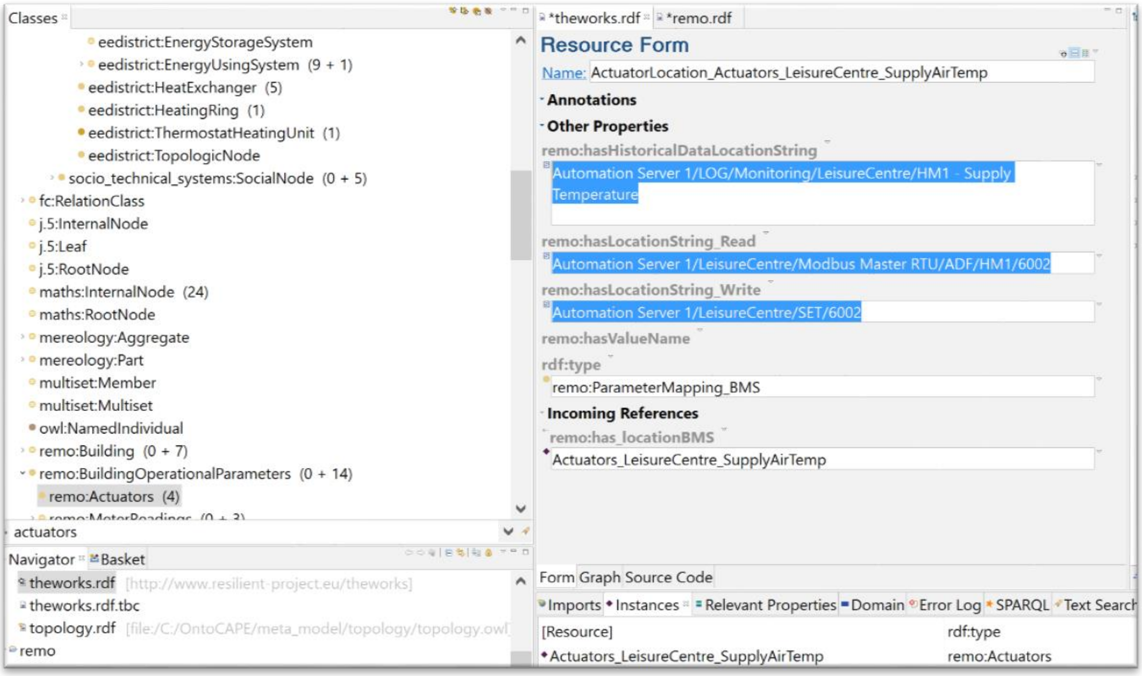


**Figure 65. The string value is given to the individual representing the endpoint location of the parameter**

### 3. Numerical value and dimension

- Certain parameters in the ontology will automatically be given an associated scalar value under the class `ScalarValueClass` using constructors, as shown in Section 6.1.2. (refer to Table 32). An example of this is shown below.
  - Referring to Figure 66 below, every newly defined property of a biomass boiler will have a scalar value individual created and associated with it. Below the scalar value for the individual 'DistanceToBiomassSupplier_BiomassBoiler_1' is shown.

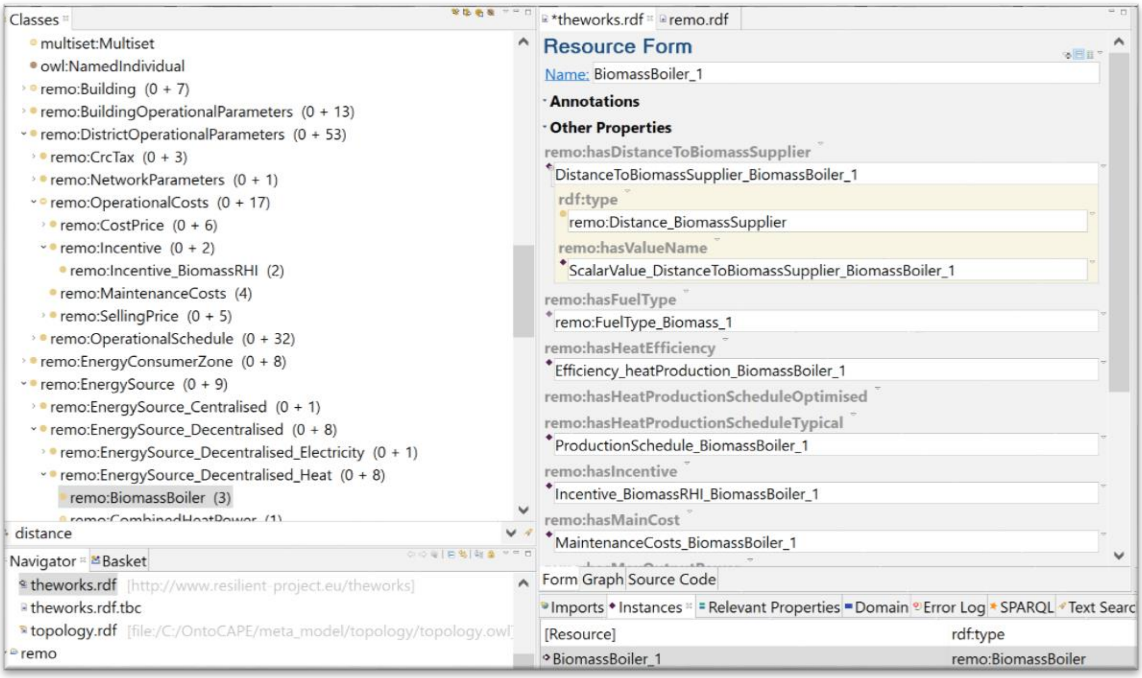



**Figure 66. Scalar value individual defined automatically for every property of biomass boiler**

- Consequently, these scalar value individuals are assigned a numerical value and dimension, as shown below in Figure 67. A list of dimensions is available for the user while assigning the dimension and these dimensions are listed from the `UnitOfMeaure` class belonging to **`system`** ontology.

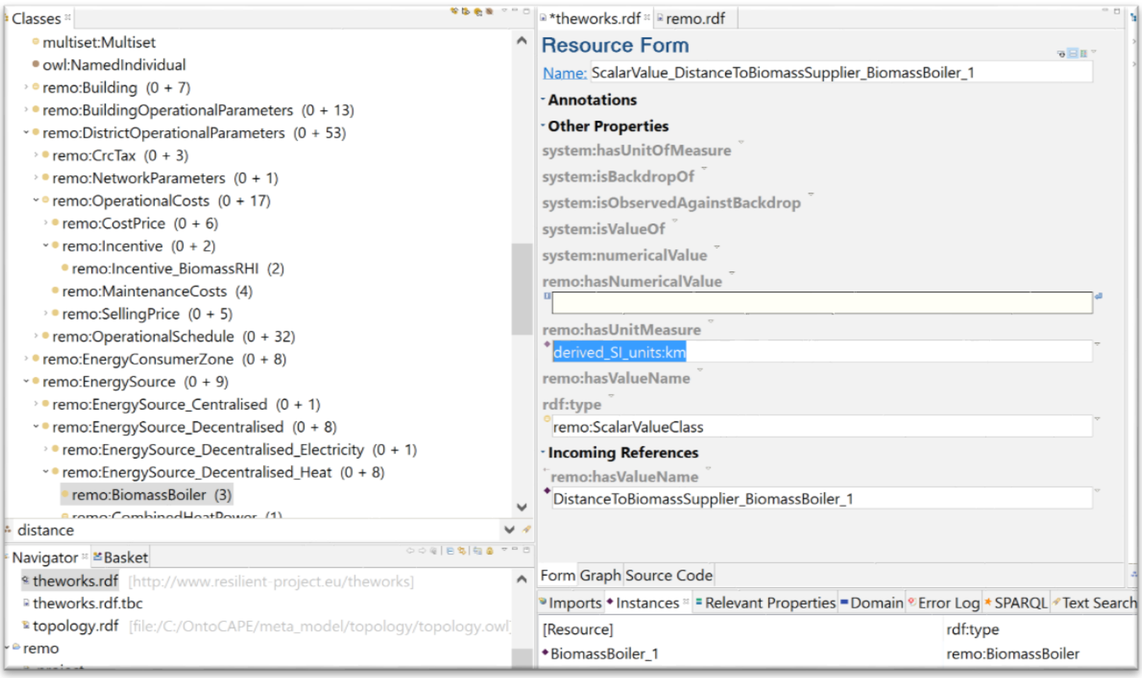



**Figure 67. Numerical value and dimension being assigned by user for every individual representing a scalar value**

### 4. Use case initialisation

Use case instantiation is the final step before reasoning the ontology where individual use cases under the `UseCases` class and its subclasses are defined. The property that assigns where the use case is applicable for is also initialised for each use case. Below, sample use cases are defined in **`theworks`** ontology. An example is provided for each category of generic use cases presented from Section 5.3. Detailed steps for each are presented below:

*Sample use case 1: Training of prediction model for the fitness room in the Leisure Centre*

- The individual under the class `UseCases_Building_Prediction_RoomAhu` is defined and named *UseCases_Building_Prediction_RoomAhu_LeisureCentre_FitnessRoom*.
- The property ***isApplicableFor*** for the newly defined individual is assigned a target area. Here, the prediction scenario is applied to the fitness room in the Leisure Centre building, as shown below in Figure 68.

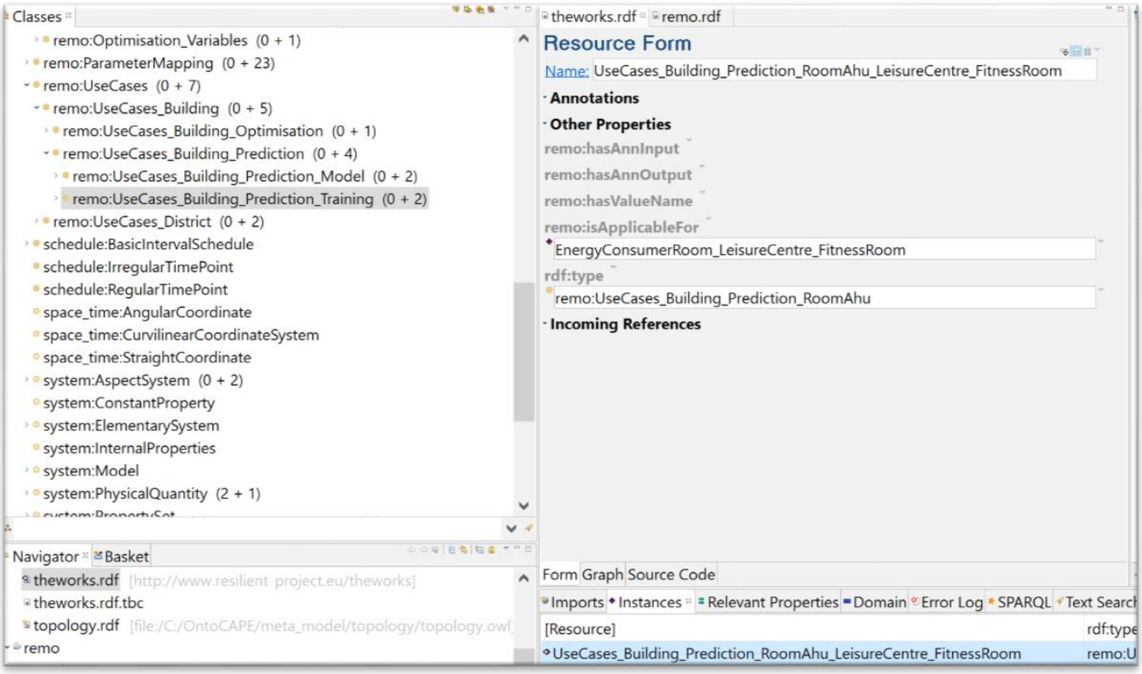



**Figure 68. Sample use case 1 is instantiated**

*Sample use case 2: Optimisation of air handling unit in fitness room*

- An individual under the class `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` is defined to represent this scenario. The individual here is named *UseCases_Building_Prediction_RoomAhu_LeisureCentre_FitnessRoom.*
- The property ***isApplicableFor*** for the newly defined individual is then assigned a target area. Here again, the fitness room is selected from the Leisure Centre, as shown below in Figure 69.

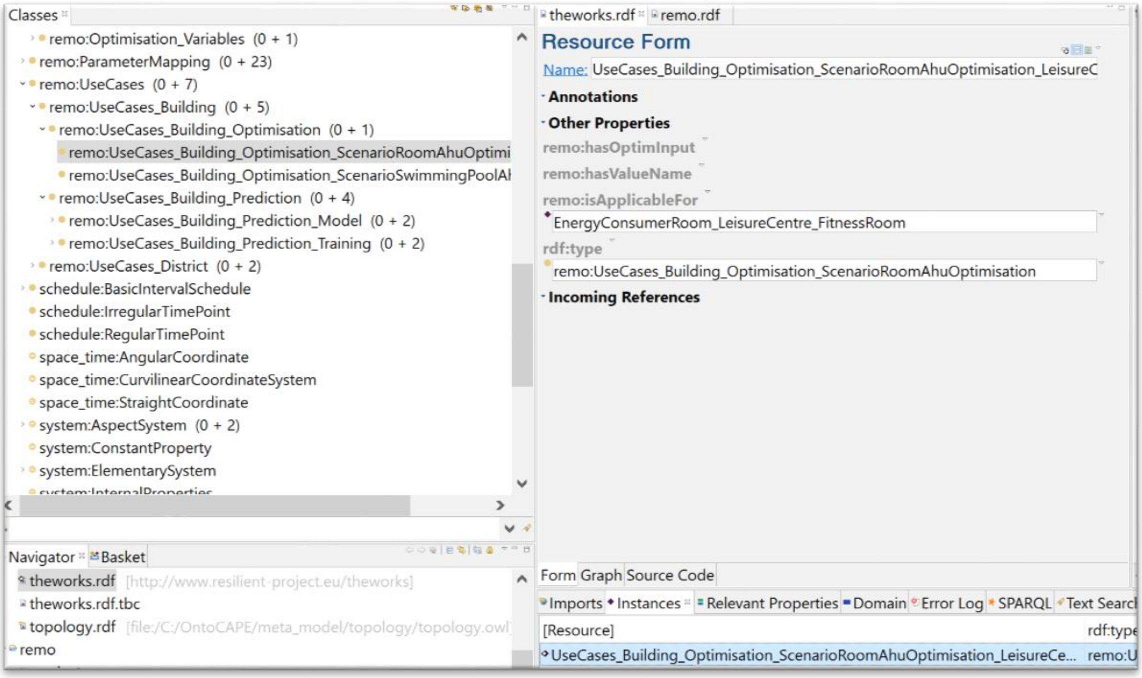



**Figure 69. Sample use case 2 is defined**

*Sample use case 3: District optimisation of Ebbw Vale using typical demand profiles of buildings*

- An individual for the class `UseCases_District_Optimisation_TypicalDemand` is instantiated. The individual here is named *UseCases_District_Optimisation_EbbwVale*. This individual represents the scenario for optimisation of the heat production schedules of the different energy sources in the district.
- The property ***isApplicableForDistrictOptimisation*** for the newly defined individual is assigned to an energy producer building from the district. The energy producer building instance selected was *EnergyConsumerBuilding_EnergyCentre*, as shown below in Figure 70.

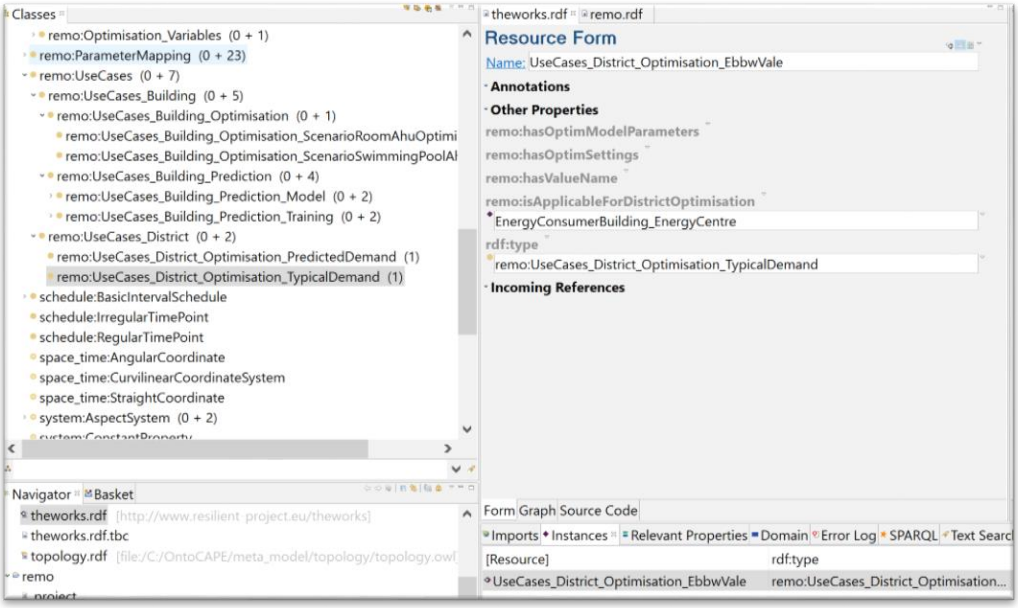


**Figure 70. Sample use case 3 is defined**

*Sample use case 4: Training of heat demand prediction model for Leisure Centre building*

- The individual under the class `UseCases_Building_Prediction_OverallDemandProfile_Heat` is defined.
- The property ***isApplicableForTotalDemandPrediction*** for the newly defined individual is assigned to an energy consumer building. Here, the energy consumer building for which the demand prediction model is required is selected. In this example case, the Leisure Centre building instance is selected – *EnergyConsumerBuilding_LeisureCentre*, as shown below in Figure 71.

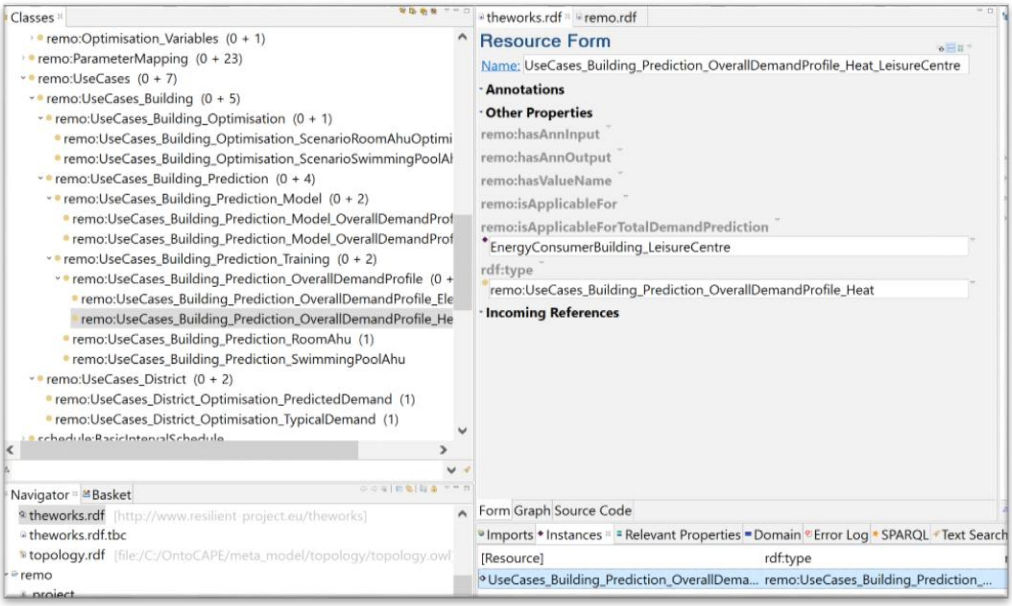


**Figure 71. Sample use case 4 is defined**

*Sample use case 5: Running of the heat demand prediction model of the Leisure Centre building.*

- An individual under the class `UseCases_Building_Prediction_Model_OverallDemandProfile_Heat` is instantiated.
- The property ***isApplicableForTotalDemandPrediction*** for the newly defined individual is assigned to an energy consumer building. In this example case, the Leisure Centre building instance is selected, as shown below in Figure 72.

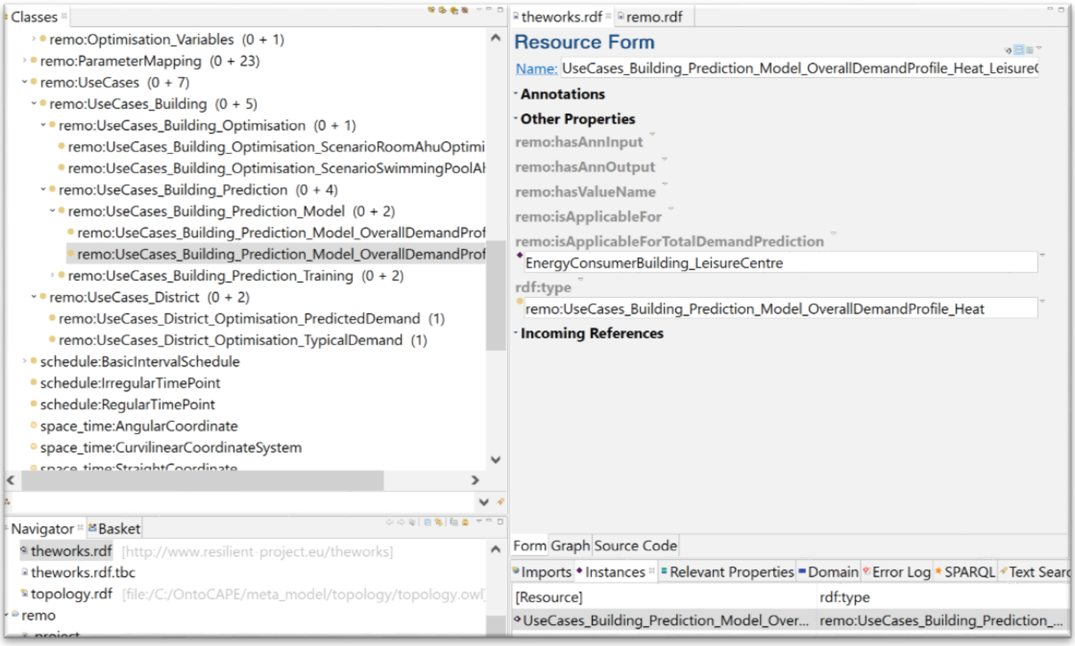


**Figure 72. Sample use case 5 is defined**

*Use case 6: District optimisation of Ebbw Vale using predicted demand profiles of buildings*

- An individual under the class `UseCases_District_Optimisation_PredictedDemand` is defined. This individual represents the scenario for optimisation of the heat production schedules of the different energy sources in the district using the predicted demand profiles of each building in the district. The district here is Ebbw Vale, and the instance is named *UseCases_District_Optimisation_PredictedDemand_EbbwVale.*
- The property ***isApplicableForDistrictOptimisation*** for the newly defined individual is assigned by the user. Here, the energy producer building from the district is selected by the user, similar to use case 3, as shown below in Figure 73.

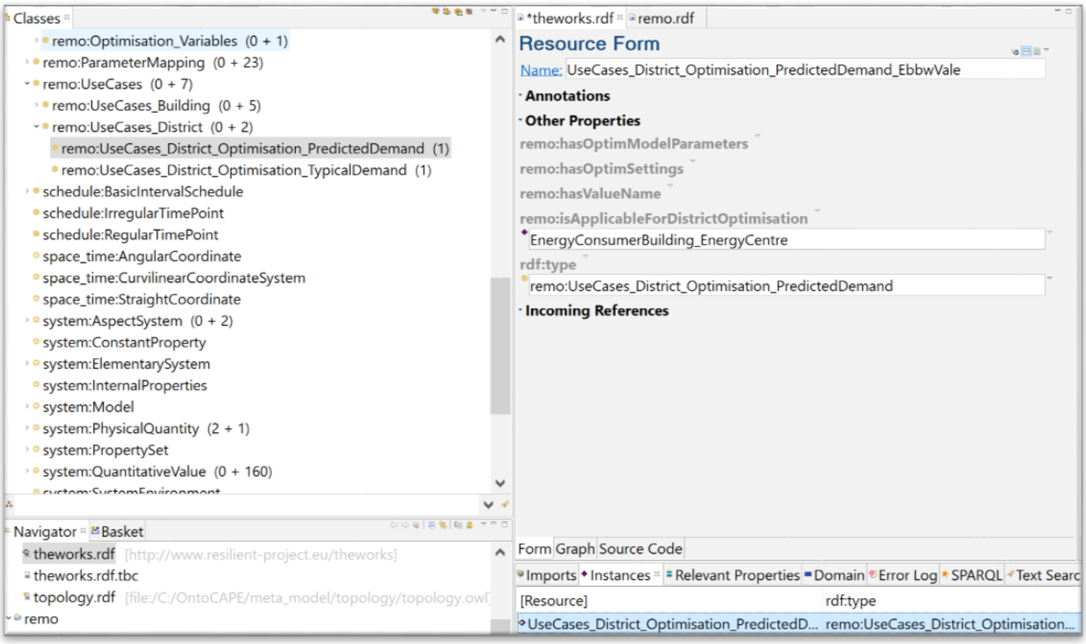



**Figure 73. Sample use case 6 is defined**

Once the instantiation is complete, the consistency of the ontology can be checked, and it is put forward for reasoning as shown in Chapter 7.

## 6.2. Implementation of functional layer

The demand side optimisation module was developed and tested in the SportE2 project, and was also implemented over the web across three pilots in the project (Petri et al. 2014b). More details about this can be found in the project document (Cardiff University 2015). The major aspects of the demand side optimisation module that needed to be implemented in the system framework for real-time energy management were:

- *ANN executable files for each scenario* – executable files can be made available through the MATLAB 2015 deploy tool[18], which was similar to the approach taken in SportE2 (Cardiff University 2015).
- *Optimisation model framework* – the optimisation framework used in SportE2 (Yang et al. 2014) was based on a general integrated optimisation design software SiPESC.OPT (Yang et al. 2011). This software also needs to be implemented in the overall system framework, which runs the optimisation models of the demand side use cases. Here, SiPESC.OPT works along with ANN executable files, as most of the building optimisation requires their respective prediction models.

[18] http://uk.mathworks.com/help/compiler_sdk/ml_code/deploytool.html

Supply side optimisation on the other hand was implemented through MATLAB. The MATLAB analytical model along with its optimisation model is exported as a standalone executable program for external deployment called District_SupplyOptim.exe. The export is made possible through the deploy tool[19] in MATLAB. The executable program needs three input files for it to run. These three input files are:

- Optimisation file with optimisation-related parameters for the NSGA-II algorithm to run (see appendices for MATLAB code).
- The analytical model file that contains all the static information for the analytical model to run (see appendices for MATLAB code).
- The demand schedules of all the buildings, which are stored in an Excel file.
- The production schedules of all the sources, which are stored in an Excel file.

This executable file can be linked to any external program. To complete the functional layer, ANN models are also needed for prediction of building heat and electricity demand profiles. Only preliminary work on these models was conducted due to lack of required data for testing. However, these models can also be similarly provided as executable files through MATLAB 2015, similar to the approach followed for SportE2.

The overall functional layer, therefore, can be detailed as shown below in Figure 74:

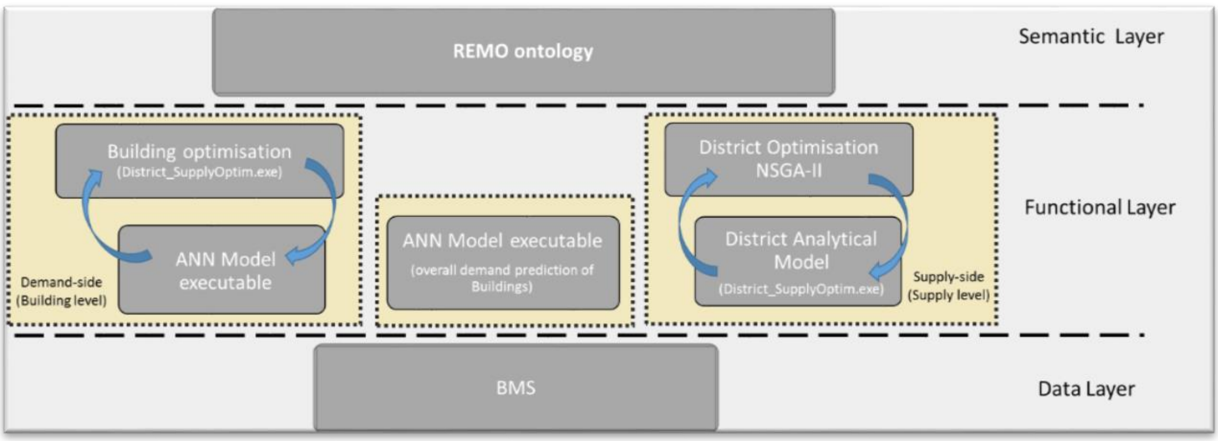



**Figure 74. Detailed functional layer of the framework**

[19] http://uk.mathworks.com/help/compiler_sdk/ml_code/deploytool.html

## 6.3. Implementation of visual layer

The Computational Urban Sustainable Platform (CUSP) is a prototype designed by the author and colleagues (Howell et al. 2016) to demonstrate a typical visual layer for the overall framework. The platform runs on the Unity Pro game engine[20], and one of the reasons for choosing this was its seamless cross-platform deployment capabilities with the efficient use of computer resources. Here, the building IFC models of the work site were placed on the actual topography of the site. These IFC models, in the future, can also be interactive models; wherein the user will be able to query information about each building's IFC model by interacting with it. A screenshot of the IFC models on CUSP is provided below in Figure 75:


**Figure 75. Screenshot of the visual interface showing 3D IFC model of buildings on the site**

The platform is envisioned to be a tool to manage future cities through an integrated semantic approach, and therefore it had to consider all domains that contribute to sustainability and not just energy. CUSP, therefore, is designed to have many dashboards, each dealing with a particular domain. The first prototype of CUSP had dashboards for energy and water domains, each capable of different functionalities.

**REMO** ontology, therefore, fits in well with the overall vision of this future cities tool, because ontologies are perfect for cross-domain integration and collaboration. The authors major contribution here was designing the energy related use cases for the CUSP energy dashboard. In this particular case, **REMO** ontology can be integrated into CUSP

[20] https://unity3d.com/unity

and can consequently help the running of the use cases of the energy dashboard. The energy dashboard once integrated with **REMO** ontology allows reuse of its applications for any site in the future, provided the site is instantiated in the **REMO** ontology. Some of the current use cases of the energy dashboard designed by the author include:

- Running of the machine-learning algorithms (ANN models) for prediction of the overall demand schedules of each building (day-ahead demand forecasts).
- Optimisation of the operational schedules of the energy sources in the district by running the district optimisation discussed in Section 4.2.2. The optimisation results provide useful insight into the management of the network and aid day-ahead decision-making for facility managers.
- The platform also allows users to calculate the key energy performance indicators of the district, with a 24-hour horizon, based on their preference of generation unit usage. These indicators are displayed in a radar graph which visualises the performance impact of the strategy chosen (see Figure 77 later). The indicators are computed by running the analytical model described in Section 4.2.2.

Figures 76 and 77 below show some of the screenshots of the first prototype developed:


**Figure 76. Screenshot of the visual interface displaying heat and electricity demand of a building in Ebbw Vale**

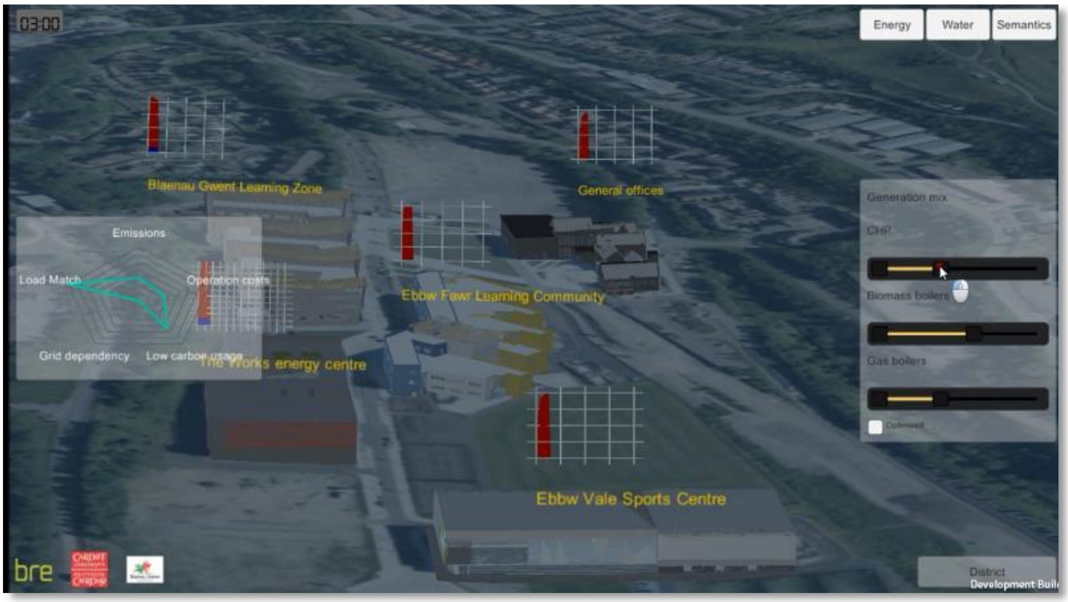



**Figure 77. Screenshot of the visual interface showing key performance indicator calculations for the chosen strategy**

Future work for CUSP which will be undertaken by author's colleagues would look into utilising a cloud-based approach to improving software performance. Further work is also needed on the CUSP district management tool, as it needs to be integrated with the web implementation of **REMO** ontology. The web implementation of the ontology would then enable the user interface to seamlessly query the ontology and retrieve information as and when needed.

Linking CUSP to a BIM server is also being investigated by the team, which will allow access to BIM information in the platform. One advantage of having BIM servers is that changes made to BIM models can be updated easily in the platform, which also makes the CUSP platform easily replicable to future sites. Semantisation of IFC model information in the platform is also supported by these BIM servers.

Usage of semantic modelling techniques to the platform is important to utilise the numerous analytics components alongside each other and to connect these to the varied data sources as well as to the other dashboards. The semantic model allows the integration of many heterogeneous data sources such as the demand schedules, meter readings, static data, IFC models and so forth. This knowledge can then be used by analytics applications such as production schedule optimisation, 3D visualisation, demand prediction and much more. Moreover, the semantic approach allows adding of further data sources and analytics with far less effort. The overall vision of the CUSP architecture within the framework of this research is presented below in Figure 78:

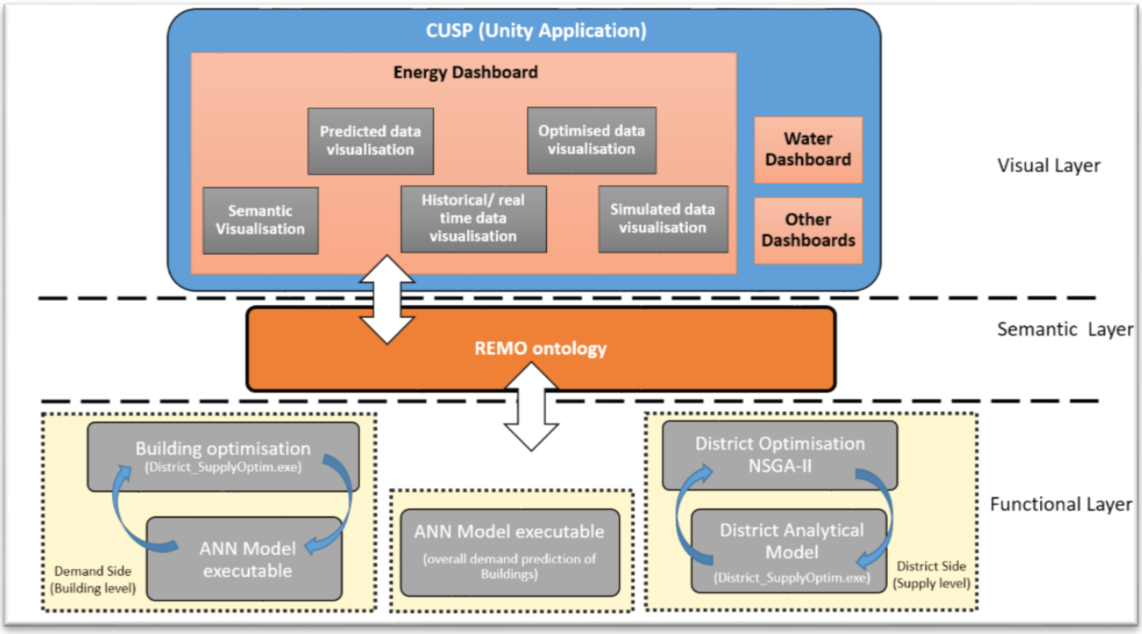



**Figure 78. CUSP architecture detailed in Visual layer**

## 6.4. Execution of use cases for the framework

A generic BPMN use case diagram showing the interaction between the various layers of the framework is shown in Figure 79 later. The User Interface (CUSP, for example) allows users to choose a particular use case through the energy dashboard. The user can select any of the following use cases:

- Use case 2 – optimisation of building-related use cases (for example, optimisation of the air-handling unit in a building).
- Use case 3 – district production schedule optimisation using typical demand profile of each building.
- Use case 5 – running of prediction models to predict overall demand of buildings (available for each building in the district).
- Use case 6 – district production schedule optimisation using optimised demand profile of each building.

Use cases 1 and 4 are not available during the operational stages once the framework is implemented because it involves training of ANN models. Algorithm for training of prediction models before using the framework for real-time operations is presented below:

```
1. Start.
2. Select use case to run.
3. Script file triggered based on use case selection.
4. Script file executes set of SPARQL queries to query the ontology.
```

5. Responses to these queries stored in the script file.
6. Script file from step 4 is used to query the BMS/EMS for historical data.
7. Responses from step 5 used to generate input file and output file using post-processing algorithms.
8. The input and output files used to train the prediction model.
9. These trained models are exported as executables to be implemented with the framework.
10. **End.**

Once the ANN models are trained and implemented in the framework, use cases 2, 3, 5, and 6 are available to be executed in real time. The algorithm for running these use cases using the framework is presented below:

1. **Start**
2. Select use case to run.
3. Script file triggered based on use case selection (in the visual layer through the user interface).
4. Script file executes set of SPARQL queries to query the ontology (in the semantic layer).
5. Responses to these queries stored in the script file.
6. Script file from step 4 used to querying the BMS/EMS (in data layer) for real-time information.
7. Responses from step 5 used to generate input file using post-processing algorithms.
8. Input file triggers mathematical model or prediction model or optimisation model (in the functional layer) based on the use case selection.
9. Running of these models and computing solutions.
10. Set solution in BMS/EMS based on user preferences.
11. **End.**

Based on the user's interaction with the user interface, the use case is selected, and a corresponding script file queries the ontology using SPARQL queries. The query returns all the relevant parameters from the ontology for the functional layer. Some of these parameters might be static and some dynamic. In the case of dynamic parameters whose value changes in real time, the ontology returns the location from where the parameter’s updated value can be read (BMS or EMS). The data layer comes into play here and provides the required data to the functional layer. Consequently, the functional layer is triggered where mathematical model/prediction model/optimisation model are run based on the use case selected. The results are displayed in the user interface, and the decision can be made here on any implementation if necessary. Any decision made is set in the BMS/EMS by using the setpoint locations of the decision variables. The details of this framework need to be worked on in future research, especially the script files that link the different layers together. Figure 79 below shows a BPMN diagram which represents the running of use cases using the framework during the operational stages.

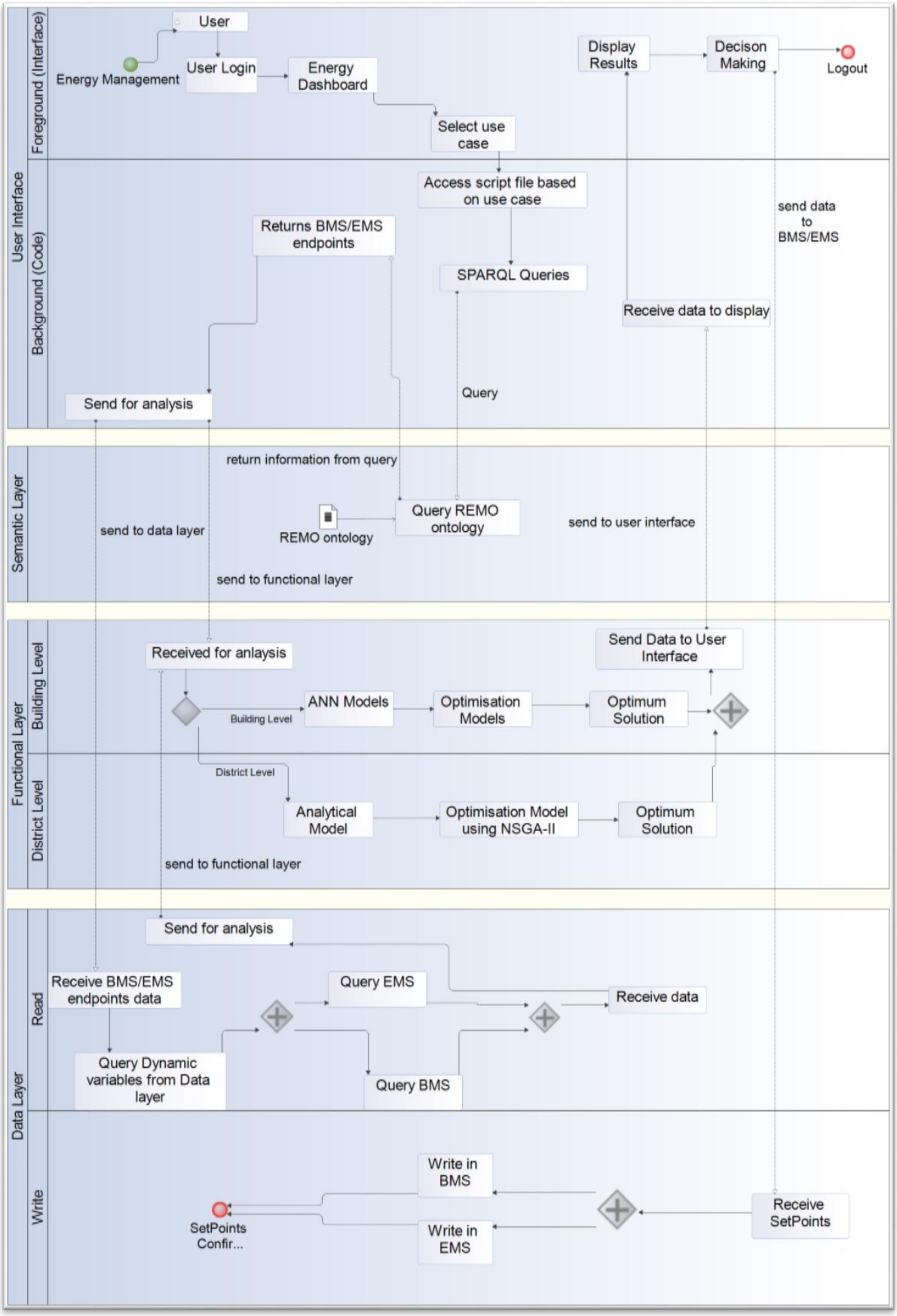



**Figure 79. Generic use case using the framework**

# 7. System Validation

Different types of validation steps are taken to evaluate the ontology as discussed in the validation methodology in Chapter 3. This chapter also provides testing and validation of the **REMO** ontology and also shows how the ontology can be used to support the running of the use cases of the framework.

## 7.1. Ontology validation

### 7.1.1. Consistency checking of REMO ontology

TopBraid allows for three types of consistency checking, all of which were implemented for **REMO** ontology and **`theworks`** ontology.

- The real-time syntax checking runs when the user is editing the ontology. Invalid statements would be outlined in red, as shown below in Figure 80:

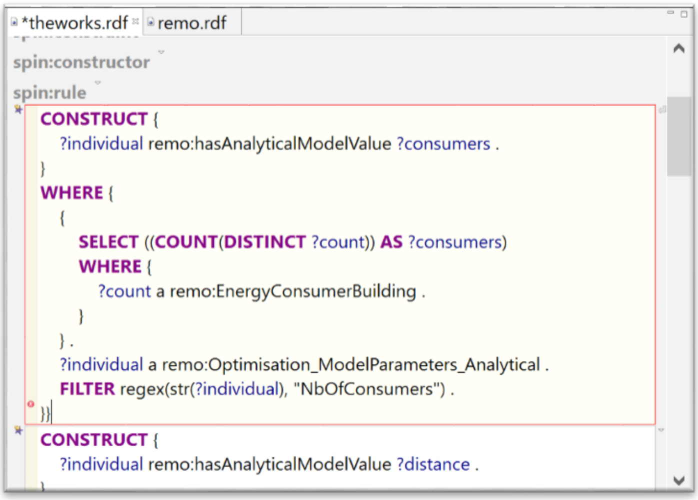



**Figure 80. Real-time syntax checking in TopBraid**

- Constraint checking in TopBraid is performed after each editing step. Thus, any violations of OWL restrictions and global range restrictions, or SPIN constraints would be displayed under the problems tab, as shown below in Figure 81.

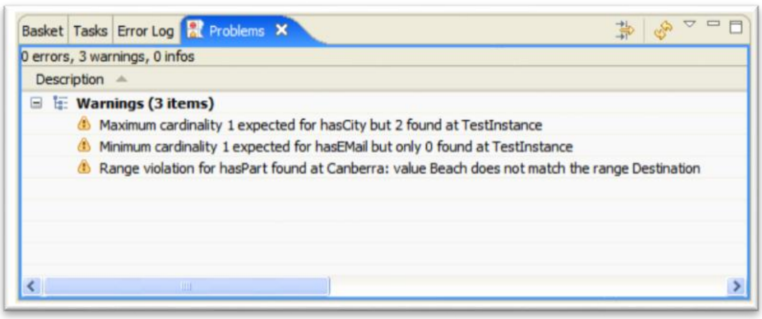



**Figure 81. Consistency checking feature in TopBraid.**

- Finally, semantic checking checks the semantic validity of the ontology which is performed by the built-in OWL DL inferencer OWLIM[21]. The semantic check is also performed while reasoning the ontology.

As TopBraid is one of the most advanced ontology editing platforms, the consistency checks were performed as **theworks** and **REMO** ontology was being built and errors were mitigated simultaneously.

### 7.1.2. Reasoning results and validation of **theworks** ontology

Various reasoning engines can be used in TopBraid. The reasoning engines used in this research were: Jena built-in reasoner and TopSPIN (SPARQL queries). Running reasoning engines on **theworks** ontology helped to infer new knowledge especially for the `UseCases` class and its subclasses because they contained the majority of the rules as described in Section 6.1.2. Reasoning knowledge was cross-checked manually by the author with the use case knowledge gained from action research, and the results of this are presented below.

*Sample use case 1 reasoning results*

Reasoning **theworks** ontology retrieves the ANN input and output-related data for the sample use case 1 defined in **theworks** ontology as shown below in Figure 82.

[21] https://www.w3.org/2001/sw/wiki/Owlim

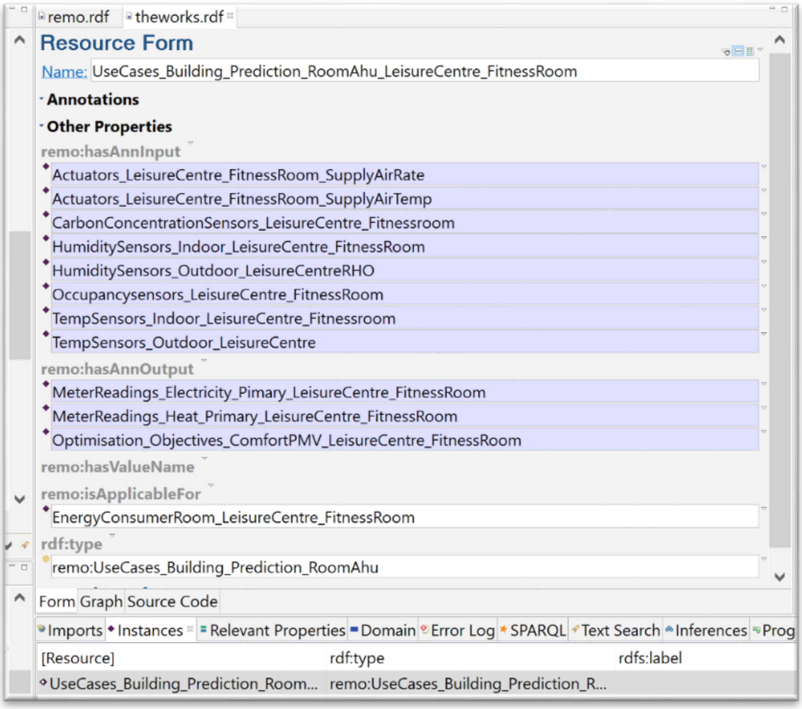



**Figure 82. Reasoning results for use case 1 defined in `theworks` ontology**

Here, the individuals highlighted in blue are inferred through the reasoning process. Table 33 below shows the results of the reasoning compared to the action research knowledge. It confirms that all the required knowledge was successfully retrieved through the reasoning process.

**Table 33. Sample use case 1 reasoning results of `theworks` ontology**

| **Use case 1** | **Training data for ANN input and output needed for AHU scenario** | |
|---|---|---|
| | **SportE2 Project Knowledge** | **Reasoned Knowledge** |
| *Input parameters* | Outdoor temperature sensor | ✓ |
| | Indoor temperature sensor | ✓ |
| | Indoor humidity sensor | ✓ |
| | Outdoor humidity sensor | ✓ |
| | Carbon concentration sensor | ✓ |
| | Actuators related to air-handling unit system | ✓ |
| | Occupancy sensors | ✓ |
| *Output parameters* | Comfort parameter | ✓ |
| | Electricity meter of room | ✓ |
| | Heat meter of room | ✓ |

Each individual inferred has associated properties linked to it. Figure 83 below shows the associated properties linked to the individual representing the carbon concentration sensor.

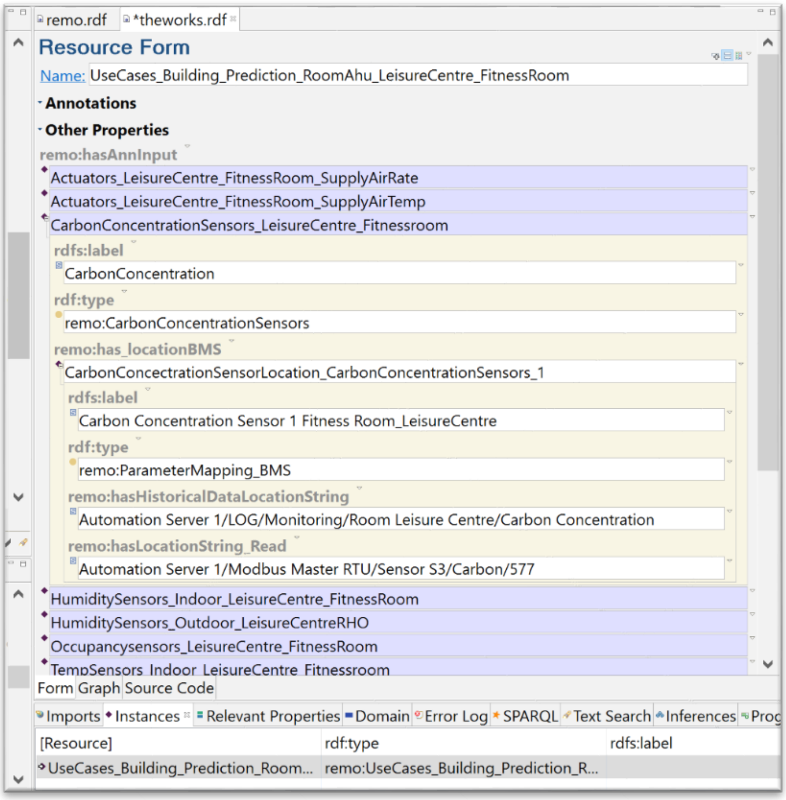



**Figure 83. Individual representing the carbon concentration sensor and its associated properties**

*<u>Sample use case 2 reasoning results</u>*

Inferencing the ontology retrieves the decision variables, optimisation model input, and optimisation objectives for the use case 2 individual defined. The results are shown in Figure 84 and Table 34.

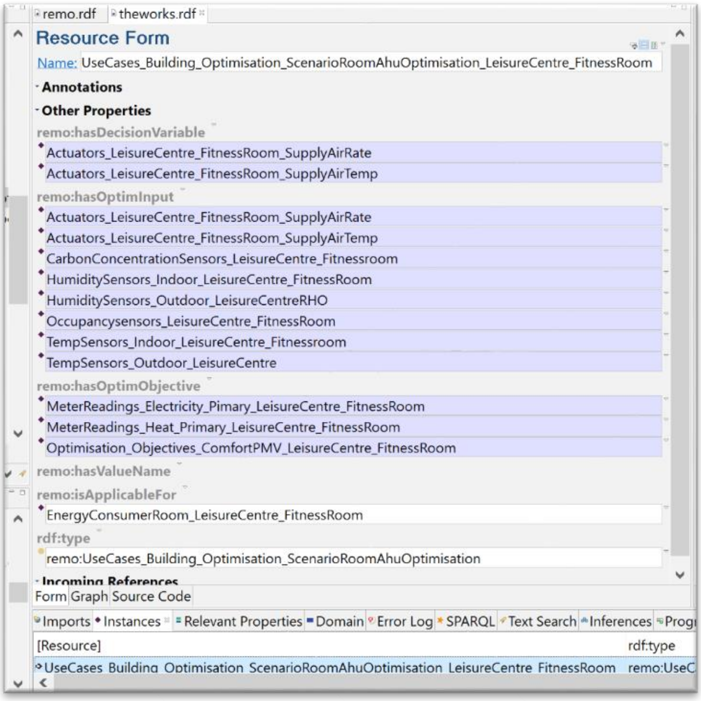



**Figure 84. Reasoning results for use case 2 of `theworks` ontology**

**Table 34. Use case 2 reasoning results**

| **Use case 2** | **Optimisation input and output data for AHU Scenario in building** | | |
|---|---|---|---|
| | | **SportE2 Project Knowledge** | **Reasoned Knowledge** |
| ***Input parameters*** | *Other input for optimisation model* | Outdoor temperature sensor | ✓ |
| | | Indoor temperature sensor | ✓ |
| | | Indoor humidity sensor | ✓ |
| | | Outdoor humidity sensor | ✓ |
| | | Carbon concentration sensor | ✓ |
| | | Occupancy sensors | ✓ |
| | *Decision variables* | Actuators related to air-handling unit system | ✓ |
| ***Output parameters*** | | Comfort parameter | ✓ |
| | | Electricity meter of room | ✓ |
| | | Heat meter of room | ✓ |

## *Sample use case 3 reasoning results*

Reasoning the ontology retrieves the decision variables, optimisation model parameters, optimisation objectives, and optimisation settings that are needed for the district optimisation model to run. Reasoning results are shown in Figure 85 and Table 35.

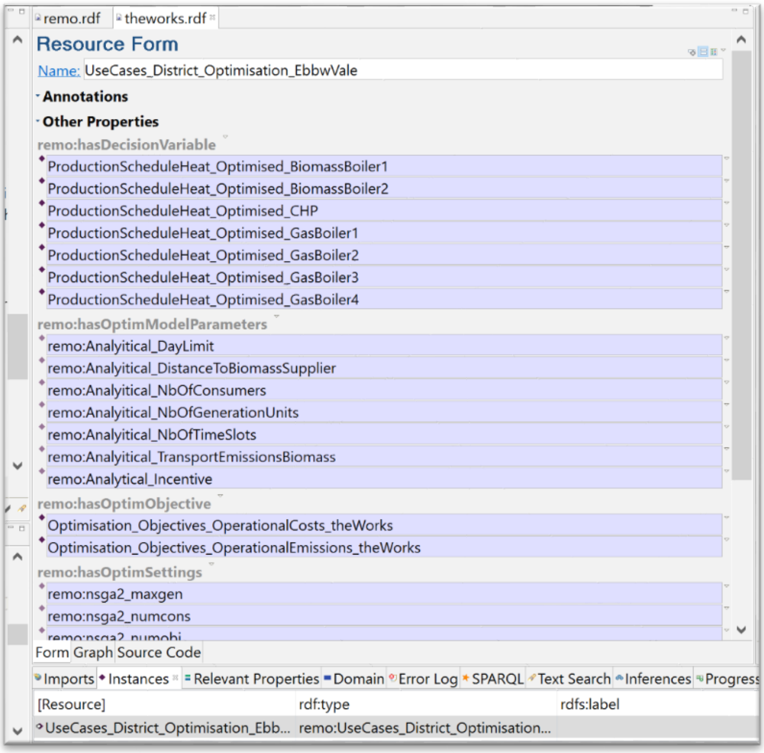



**Figure 85. Reasoning results**

The numerical values of the various analytical model parameters are also inferred through reasoning. Figure 86 below shows an example, where the ontology infers that the number of consumers in the district relying on district heating system is 6.

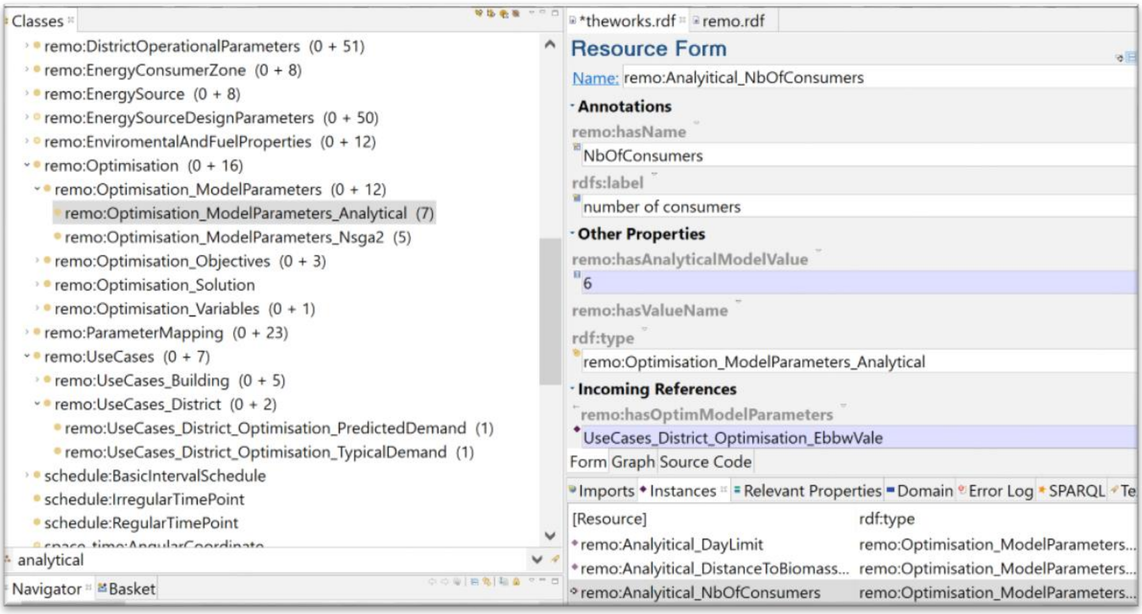



**Figure 86. The reasoning of numerical values of the analytical model parameters**

**Table 35. Use case 3 reasoning results**

<table>
<tr><th>Use case 3</th><th colspan="3">Input and output data for optimisation of production schedules of the various energy sources in the district</th></tr>
<tr><td></td><td></td><td>Knowledge from District Optimisation Methodology</td><td>Reasoned Knowledge</td></tr>
<tr><td rowspan="3">Input parameters</td><td rowspan="2">Other input for optimisation model</td><td>NSGA-II optimisation settings related parameters</td><td>✓</td></tr>
<tr><td>District analytical model parameters</td><td>Partly available</td></tr>
<tr><td>Decision variables</td><td>Energy sources heat production schedule</td><td>✓</td></tr>
<tr><td rowspan="2">Output parameters</td><td rowspan="2"></td><td>Cost parameter</td><td>✓</td></tr>
<tr><td>Emission parameter</td><td>✓</td></tr>
</table>

In the case of district optimisation, many other related parameters are indirectly needed for the district optimisation module to run. These are not directly reasoned and would have to be queried against the ontology separately with a set of pre-defined SPARQL queries, some of which are shown in Section 7.1.3.

*Sample use case 4 reasoning results*

Reasoning **`theworks`** ontology retrieves the ANN input and output-related data for training of the ANN model which is needed for the overall building demand prediction, as shown below in Figure 87 and Table 36.

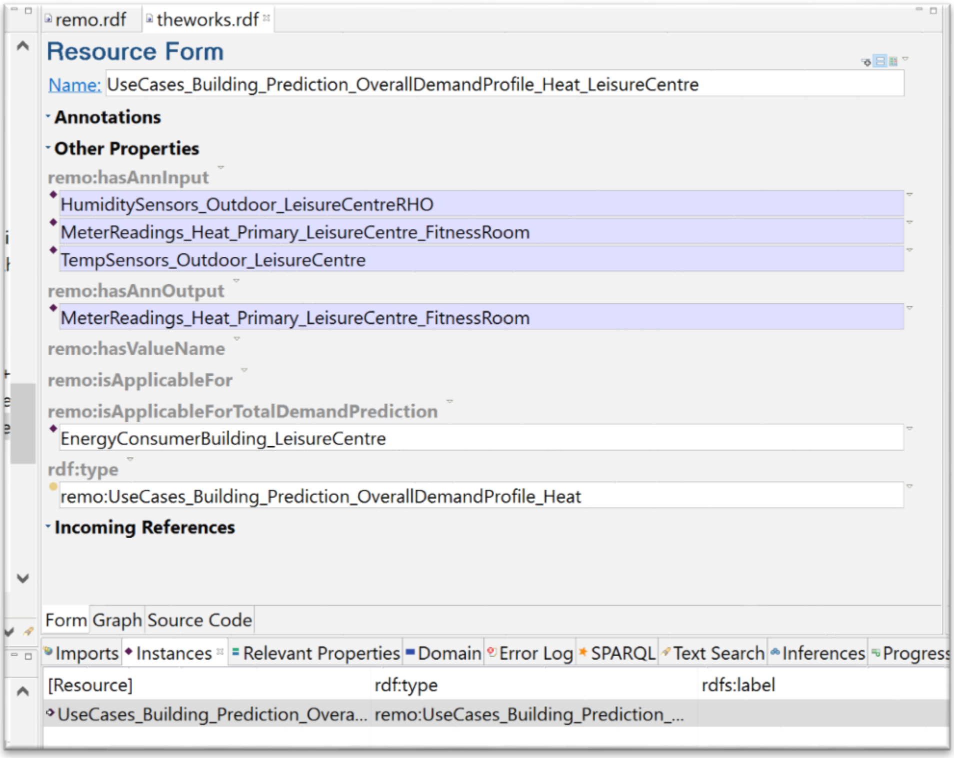



**Figure 87. Reasoning results for use case 4 of `theworks` ontology**

**Table 36. Use case 4 reasoning results**

| **Use case 4** | **ANN-related data for training the demand forecast models of individual buildings** | |
|---|---|---|
| | **Research Knowledge** | **Reasoned Knowledge** |
| *Input parameters* | Primary heat meter for building* | ✓ |
| | Outdoor humidity data | ✓ |
| | Outdoor temperature data | ✓ |
| *Output parameters* | Primary heat meter for building* | ✓ |

*Primary heat meter for the building is listed as an input and an output, because, once the data is retrieved from the BMS or EMS, different datasets (according to timestamp) are used as input and output for training purposes.

*Sample use case 5 reasoning results*

Reasoning the ontology infers the ANN input and output parameters needed for running the prediction model in this use case, as shown below in Figure 88 and Table 37:

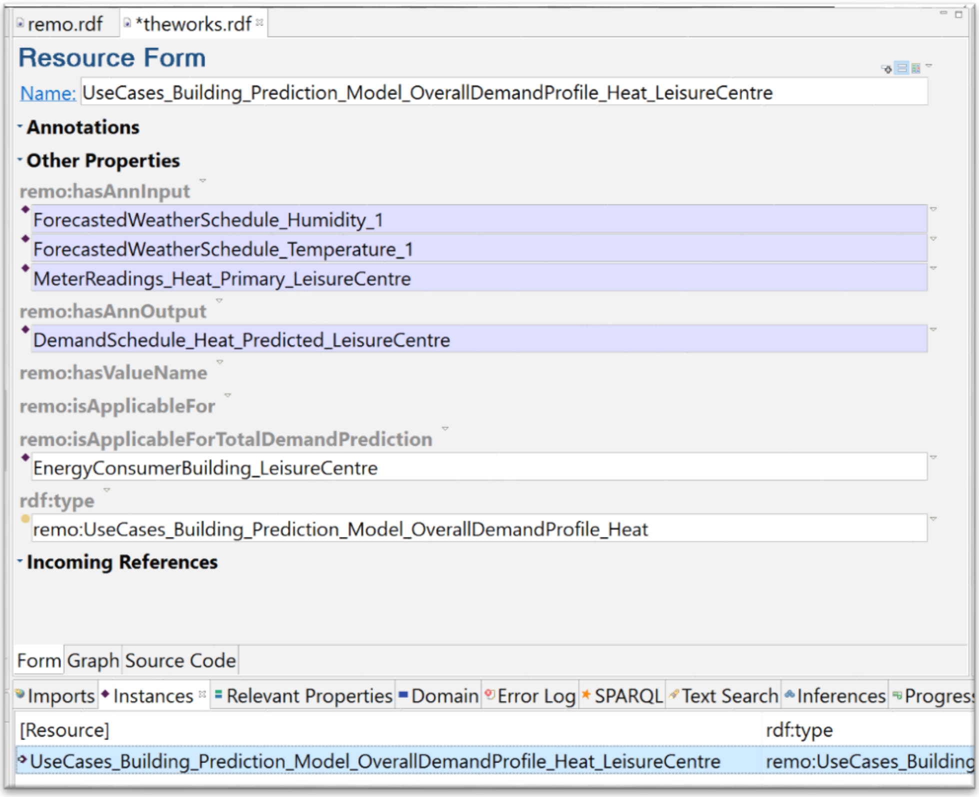



**Figure 88. Reasoning results of use case 5 of `theworks` ontology**

**Table 37. Use case 5 reasoning results**

| **Use case 5** | **ANN-related data for running of the demand forecast models of individual buildings** | |
|---|---|---|
| | **Research Knowledge** | **Reasoned Knowledge** |
| *Input parameters* | Primary heat meter for building (to retrieve previous day demand) | ✓ |
| | Forecasted outdoor temperature | ✓ |
| | Forecasted outdoor humidity | ✓ |
| *Output parameters* | Demand schedule of building (forecasted demand) | ✓ |

Reasoning results for sample use case 6 is similar to use case 3 results as presented above.

### 7.1.3. SPARQL Query validation

SPARQL queries can be used to query the ontology, and this can be used to validate the competency questions listed in Section 6.1.1 under Chapter 6. The summary of results from running the competency questions is shown below in the following tables:

**Table 38. Optimisation-related competency questions and answers**

<table>
<tr><th colspan="2">Optimisation-related questions</th></tr>
<tr><td>Question</td><td>What are the various optimisation input parameters given a <use case> :<br>< UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation ></td></tr>
<tr><td>SPARQL query</td><td>SELECT ?b ?label<br>WHERE {<br>?a remo:hasOptimInput ?b .<br>?a a remo:UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation .<br>?b rdfs:label ?label }</td></tr>
<tr><td>Response</td><td><table>
<tr><th>[b]</th><th>[label]</th></tr>
<tr><td>Actuators_LeisureCentre_FitnessRoom_SupplyAirRate</td><td>Actuator Supply Air Flow rate</td></tr>
<tr><td>Actuators_LeisureCentre_FitnessRoom_SupplyAirTemp</td><td>Actuator Supply Air Temp</td></tr>
<tr><td>CarbonConcentrationSensors_LeisureCentre_Fitnessroom</td><td>CarbonConcentration</td></tr>
<tr><td>HumiditySensors_Indoor_LeisureCentre_FitnessRoom</td><td>Indoor Humidity Sensor</td></tr>
<tr><td>HumiditySensors_Outdoor_LeisureCentreRHO</td><td>Outdoor humidity Sensor</td></tr>
<tr><td>OccupancySensors_LeisureCentre_FitnessRoom</td><td>Occupancy Sensor</td></tr>
<tr><td>TempSensors_Indoor_LeisureCentre_Fitnessroom</td><td>Indoor Temperature Sensor</td></tr>
<tr><td>TempSensors_Outdoor_LeisureCentre</td><td>Outdoor Temperature</td></tr>
</table></td></tr>
<tr><td>Question</td><td>What are the optimisation related settings for district optimisation given a <use case>:<br><UseCases_District_Optimisation_TypicalDemand></td></tr>
<tr><td>SPARQL query</td><td>SELECT ?nsga2 ?label<br>WHERE {<br>?a remo:hasOptimSettings ?nsga2 .<br>?a a remo:UseCases_District_Optimisation_TypicalDemand .<br>?nsga2 rdfs:label ?label .<br>}</td></tr>
<tr><td>Response</td><td><table>
<tr><th>[nsga2]</th><th>[label]</th></tr>
<tr><td>remo:nsga2_maxgen</td><td>maximum generations</td></tr>
<tr><td>remo:nsga2_numcons</td><td>number of constraints</td></tr>
<tr><td>remo:nsga2_numobj</td><td>number of objectives</td></tr>
<tr><td>remo:nsga2_numvar</td><td>number of variables</td></tr>
<tr><td>remo:nsga2_popsize</td><td>population size</td></tr>
</table></td></tr>
</table>

**Table 39. Prediction-related competency questions and answers**

<table>
<tr><th colspan="3">Prediction-related questions</th></tr>
<tr><td>Question</td><td colspan="2">What are the various ANN outputs needed for training the ANN given a <use case>:<br><UseCases_Building_Prediction_Model_OverallDemandProfile_Heat></td></tr>
<tr><td>SPARQL query</td><td colspan="2">SELECT ?b ?label<br>WHERE {<br>?a remo:hasAnnOutput ?b .<br>?a a remo:UseCases_Building_Prediction_RoomAhu .<br>?b rdfs:label ?label }</td></tr>
<tr><td rowspan="2">Response</td><td>[b]</td><td>[label]</td></tr>
<tr><td>MeterReadings_Heat_Primary_LeisureCentre</td><td>heat meter reading primary Leisure Centre</td></tr>
<tr><td>Question</td><td colspan="2">What are the various ANN inputs needed for running the ANN model for overall building demand given a <use case>:<br>< UseCases_Building_Prediction_Model_OverallDemandProfile_Heat ></td></tr>
<tr><td>SPARQL query</td><td colspan="2">SELECT ?b ?label<br>WHERE {<br>?a remo:hasAnnInput ?b .<br>?a a remo:UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .<br>?b rdfs:label ?label }</td></tr>
<tr><td rowspan="4">Response</td><td>[b]</td><td>[label]</td></tr>
<tr><td>ForecastedWeatherSchedule_Humidity_EbbwVale</td><td>forecasted humidity schedule</td></tr>
<tr><td>ForecastedWeatherSchedule_Temperature_EbbwVale</td><td>forecasted temperature schedule</td></tr>
<tr><td>MeterReadings_Heat_Primary_LeisureCentre</td><td>heat meter reading primary Leisure Centre</td></tr>
</table>

**Table 40. District static topology-related competency questions and answers**

<table>
<tr><th colspan="2">District-related questions – static topology</th></tr>
<tr><td>Question</td><td>List the energy sources in the energy producer building given a <energy producer building>:<br><EnergyConsumerBuilding_EnergyCentre></td></tr>
<tr><td>SPARQL query</td><td>SELECT ?b ?label<br>WHERE {<br>?a remo:includesHeatSource ?b .<br>?a a remo:EnergyProducerBuilding .<br>FILTER regex(str(?a), "EnergyConsumerBuilding_EnergyCentre") .<br>?b rdfs:label ?label<br>}</td></tr>
<tr><td>Response</td><td><table>
<tr><th>[b]</th><th>[label]</th></tr>
<tr><td>BiomassBoiler_EnergyCentre1</td><td>biomass boiler</td></tr>
<tr><td>BiomassBoiler_EnergyCentre2</td><td>biomass boiler</td></tr>
<tr><td>CombinedHeatPower_EnergyCentreCHP</td><td>Chp</td></tr>
<tr><td>GasBoiler_EnergyCentre1</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre2</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre3</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre4</td><td>Boiler</td></tr>
</table></td></tr>
<tr><td>Question</td><td>List the energy sources that supply heat given a <energy consuming building>:<br>< EnergyConsumerBuilding_LeisureCentre ></td></tr>
<tr><td>SPARQL query</td><td>SELECT ?b ?label<br>WHERE {<br>?a remo:hasHeatSource ?b .<br>?a a remo:EnergyConsumerBuilding .<br>FILTER regex(str(?a), "EnergyConsumerBuilding_LeisureCentre") .<br>?b rdfs:label ?label<br>}</td></tr>
<tr><td>Response</td><td><table>
<tr><th>[b]</th><th>[label]</th></tr>
<tr><td>BiomassBoiler_EnergyCentre1</td><td>biomass boiler</td></tr>
<tr><td>BiomassBoiler_EnergyCentre2</td><td>biomass boiler</td></tr>
<tr><td>CombinedHeatPower_EnergyCentreCHP</td><td>Chp</td></tr>
<tr><td>GasBoiler_EnergyCentre1</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre2</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre3</td><td>Boiler</td></tr>
<tr><td>GasBoiler_EnergyCentre4</td><td>Boiler</td></tr>
</table></td></tr>
</table>

**Table 41. Competency questions and answers related to energy sources and fuel properties**

<table>
<tr><th colspan="4">Energy Sources and fuel properties-related questions – static information</th></tr>
<tr><td>Question</td><td colspan="3">List the maximum or minimum output power given a <energy source>:<br>< BiomassBoiler_EnergyCentre1></td></tr>
<tr><td>SPARQL query</td><td colspan="3">SELECT ?upperBound ?name ?source<br>WHERE {<br>?n remo:hasNumericalValue ?upperBound .<br>?n a remo:ScalarValueClass .<br>?name remo:hasValueName ?n .<br>?name a remo:OutputPower_Max .<br>?source remo:hasMaxOutputPower ?name<br>FILTER regex(str(?source), "BiomassBoiler_EnergyCentre1")<br>}</td></tr>
<tr><td rowspan="2">Response</td><td>[upperBound]</td><td>[name]</td><td>[source]</td></tr>
<tr><td>495</td><td>Bounds_Upper_Biomass1</td><td>BiomassBoiler_EnergyCentre1</td></tr>
<tr><td>Question</td><td colspan="3">List the maintenance cost given a <energy source>:<br>< CombinedHeatPower_EnergyCentreCHP ></td></tr>
<tr><td>SPARQL query</td><td colspan="3">SELECT ?main_cost ?name ?source<br>WHERE {<br>?n remo:hasNumericalValue ?main_cost .<br>?n a remo:ScalarValueClass .<br>?name remo:hasValueName ?n .<br>?name a remo:MaintenanceCosts .<br>?source remo:hasMainCost ?name<br>FILTER regex(str(?source), "CombinedHeatPower_EnergyCentreCHP")<br>}</td></tr>
<tr><td rowspan="2">Response</td><td>[main_cost]</td><td>[name]</td><td>[source]</td></tr>
<tr><td>0.0043</td><td>MaintenanceCosts_Combined<br>HeatPower_1</td><td>CombinedHeatPower_EnergyCentre<br>CHP</td></tr>
</table>

**Table 42. Competency questions and answers related to numerical values and dimensions**

| **Numerical values and dimensions-related questions** | | |
|---|---|---|
| **Question** | List the numerical value given <scalar value name>:<br>< CHP_MaintenanceCost > | |
| **SPARQL query** | SELECT ?main_cost ?scalarValueName<br>WHERE {<br>?scalarValueName remo:hasNumericalValue ?main_cost .<br>?scalarValueName a remo:ScalarValueClass .<br>FILTER regex(str(?scalarValueName), "CHP_MaintenanceCost")<br>} | |
| **Response** | **[main_cost]** | **[scalarValueName]** |
| | 0.0043 | CHP_MaintenanceCost |
| **Question** | List the dimension given <scalar value name>:<br>< CHP_MaintenanceCost > | |
| **SPARQL query** | SELECT ?dimension ?scalarValueName<br>WHERE {<br>?scalarValueName remo:hasUnitMeasure ?dimension .<br>?scalarValueName a remo:ScalarValueClass .<br>FILTER regex(str(?scalarValueName), "CHP_MaintenanceCost")<br>} | |
| **Response** | **[dimension]** | **[scalarValueName]** |
| | derived_SI_units:EUR_per_kWh | CHP_MaintenanceCost |

**Table 43. Competency questions and answers related to dynamic information and parameter mapping**

| Dynamic information and parameter mapping-related questions | |
|---|---|
| **Question** | List all sensors given <room>:<br><EnergyConsumerRoom_LeisureCentre_FitnessRoom> |
| **SPARQL query** | **SELECT** ?sensor<br>    **WHERE** {<br>        ?room remo:hasSensors ?sensor .<br>        **FILTER** regex(str(?room),<br>"EnergyConsumerRoom_LeisureCentre_FitnessRoom")<br>    } |
| **Response** | **[sensor]** |
| | CarbonConcentrationSensors_LeisureCentre_Fitnessroom |
| | HumiditySensors_Indoor_LeisureCentre_FitnessRoom |
| | HumiditySensors_Outdoor_LeisureCentreRHO |
| | OccupancySensors_LeisureCentre_FitnessRoom |
| | TempSensors_Indoor_LeisureCentre_Fitnessroom |
| | TempSensors_Outdoor_LeisureCentre |
| **Question** | What are the optimisation related settings for district optimisation given a <use case>:<br><UseCases_District_Optimisation_TypicalDemand> |
| **SPARQL query** | **SELECT** ?actuator<br>    **WHERE** {<br>        ?energySystem remo:hasActuators ?actuator .<br>        **FILTER** regex(str(?energySystem),<br>"AirHandlingUnit_LeisureCentre_FitnessRoom")<br>    } |
| **Response** | **[actuator]** |
| | Actuators_LeisureCentre_FitnessRoom_SupplyAirRate |
| | Actuators_LeisureCentre_FitnessRoom_SupplyAirTemp |

After reasoning, the instantiated ontology, in this case, **theworks** ontology is further queried using SPARQL queries to retrieve information from the ontology needed for the numerical and optimisation models to run. The running of SPARQL queries and the information retrieved for the sample use cases are shown below:

### Sample use case 1 – Training of prediction model for the fitness room in the Leisure Centre

- Endpoint locations of historical data and their given labels can be retrieved for all the ANN input parameters relevant to this use case by using a SPARQL query, as shown below in Figure 89:

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnInput ?b .
?a a remo:UseCases_Building_Prediction_RoomAhu .
?b remo: has_locationBMS ?c .
?c remo:hasHistoricalDataLocationString ?locationID.
        ?b rdfs:label ?label }
```

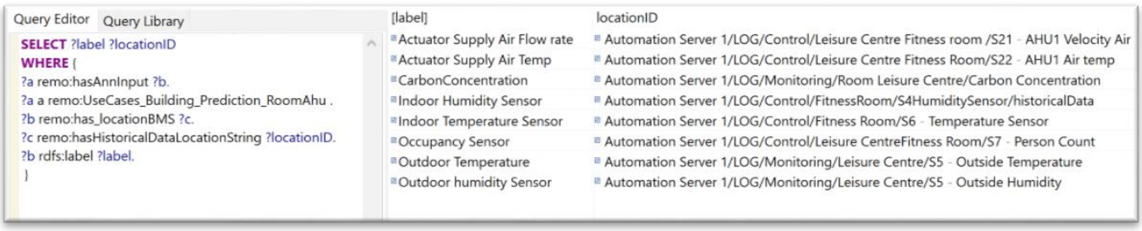



**Figure 89. SPARQL query to retrieve ANN input for use case 1 and its results**

- Similarly, ANN output parameters can also be queried using the query below (refer to Figure 90):

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnOutput ?b .
?a a remo:UseCases_Building_Prediction_RoomAhu .
?b remo:has_locationBMS ?c .
?c remo:hasHistoricalDataLocationString ?locationID.
        ?b rdfs:label ?label }
```

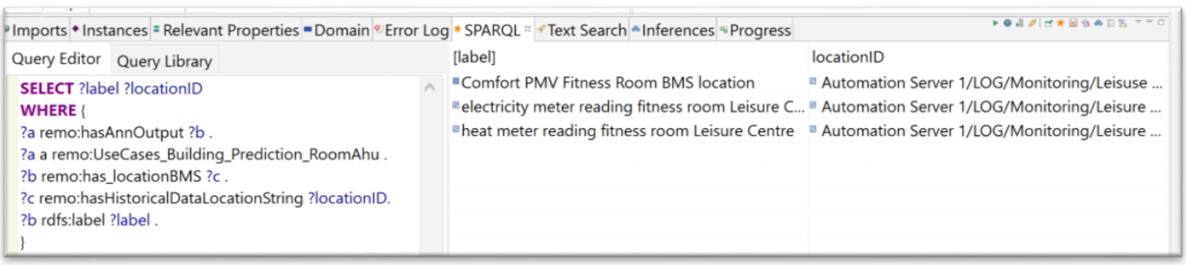



**Figure 90. SPARQL query to retrieve ANN output for use case 1 and its results**

- The results of the query are then used to retrieve data which is consequently used to train the ANN model of this use case as shown in Figure 91 below. Once trained, the ANN model can be stored as an executable file as followed in the SportE2 project, and it can be used by the optimisation modules.

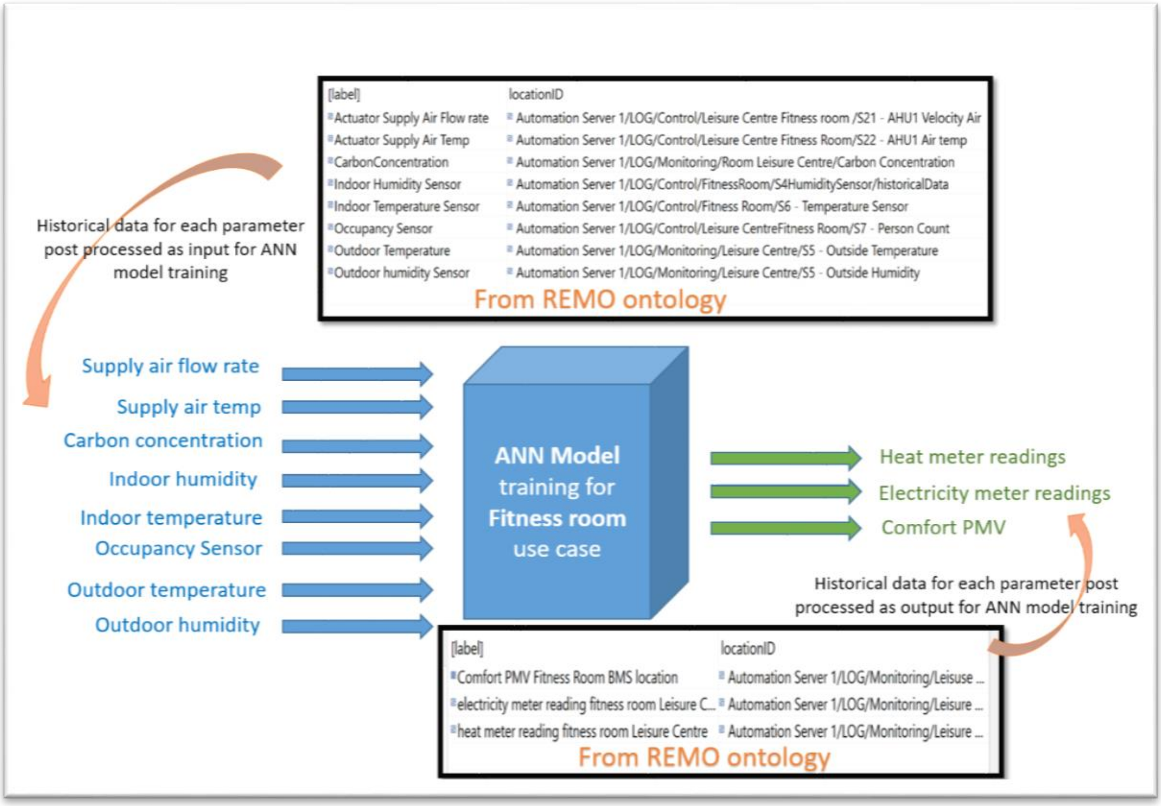



**Figure 91. Results of SPARQL query being used to train ANN model for use case 1**

### Sample use case 2 - Optimisation of air-handling unit in the fitness room

- This use case is applied in real time, and the ‘label’ and endpoint location (to retrieve the dynamic value) for each optimisation-related input parameter for this use case can be queried. The ‘label’ is used for matching the input against the input of the ANN model, which is the cost function of the optimisation. Input parameters are queried using the SPARQL query, as shown below:

```
SELECT ?label ?locationID
WHERE {
?a remo:hasOptimInput ?b .
?b remo:has_locationBMS ?c .
?c remo:hasLocationString_Read  ?locationID.
        ?b rdfs:label ?label }
```

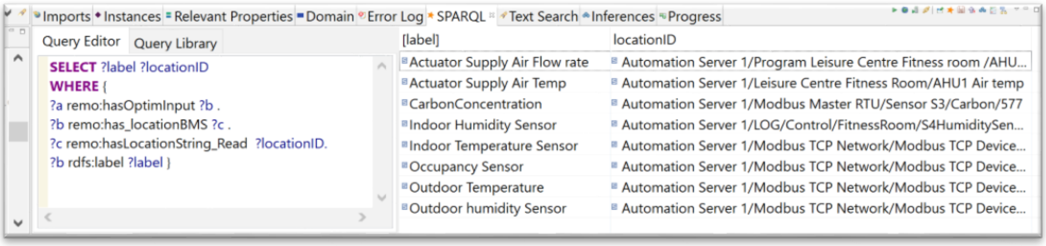



**Figure 92. SPARQL query to retrieve ANN input for use case 2 and its results**

- The dynamic information retrieved is fed into the optimisation model, as shown below in Figure 94.
- The decision variables are then optimised using methodology adopted in the SportE2 project (refer to Section 4.1). The optimised decision variables can then be set in the BMS. To do this the location of the actuator setpoint is queried using SPARQL, as shown below:

```
SELECT ?label ?setpointLocationID
WHERE {
?a remo:hasDecisionVariable ?b .
?b remo:has_locationBMS ?c .
?b a remo:Actuators.
?b rdfs:label ?label .
?c remo:hasLocationString_Write ?setpointLocationID.
        }
```

The query above gives the setpoint location in the BMS and its label in the ontology, as shown below in Figure 93:

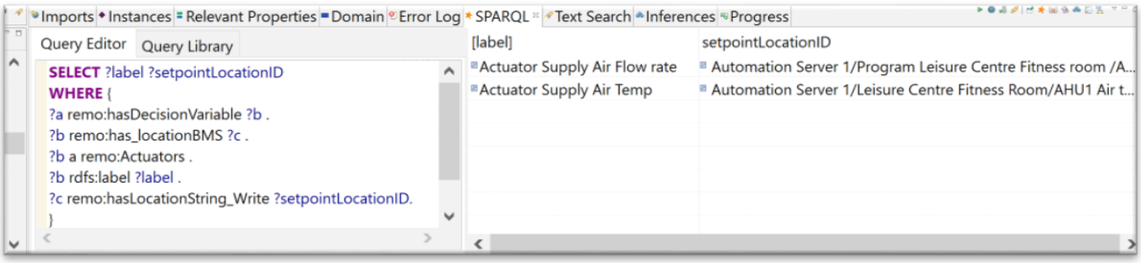



**Figure 93. SPARQL query to retrieve setpoint location of decision variables in use case 2 and its results**

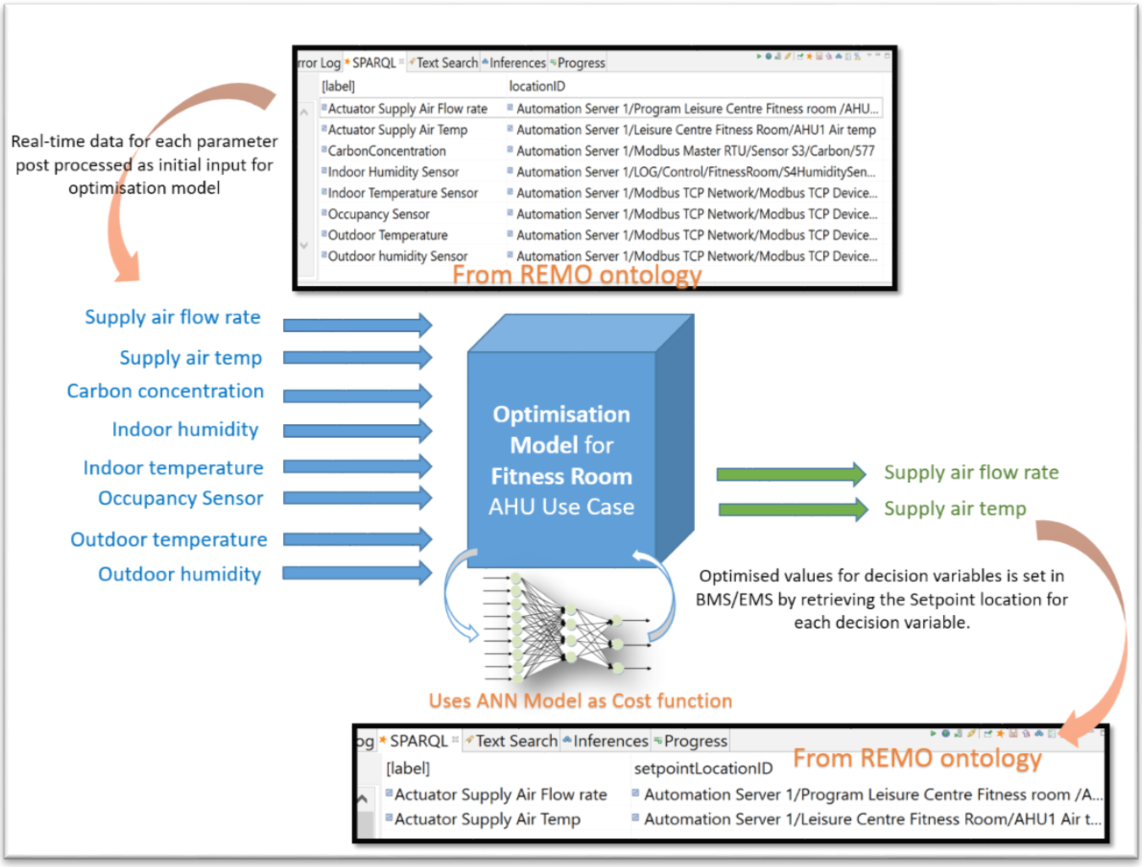



**Figure 94. Real-time data retrieved from ontology used for the optimisation model of use case 2.**

## Sample use case 3 – District optimisation of Ebbw Vale using typical demand profiles of buildings

- SPARQL queries help retrieve static parameters and their labels using the query below. This is consequently used by the district analytical model.

```
SELECT ?label ?analyticalValue
WHERE {
?a remo:hasOptimModelParameters ?b .
?a a remo:UseCases_District_Optimisation_TypicalDemand .
?b remo:hasAnalyticalModelValue ?analyticalValue  .
?b a remo:Optimisation_ModelParameters_Analytical .
?b rdfs:label ?label .
        }
```

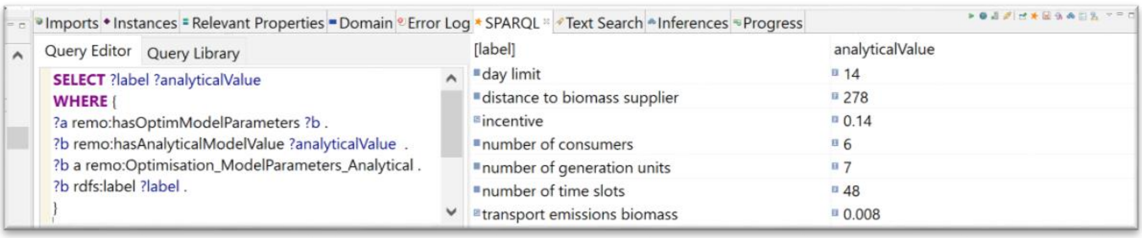



**Figure 95. SPARQL query to retrieve numerical values of district analytical model parameters**

- The individuals under the class `Optimisation_ModelParameters_Nsga2` represent various settings for the optimisation using the NSGA-II algorithm which is used for the district optimisation. The values of these parameters and their labels can be queried through SPARQL query language, as shown below:

```
SELECT ?label ?settingValue
WHERE {
?a remo:hasOptimSettings ?nsga2 .
?a a remo:UseCases_District_Optimisation_TypicalDemand .
?nsga2 remo:hasValueName ?value  .
?value remo:hasNumericalValue ?settingValue .
?nsga2 rdfs:label ?label .
                }
```

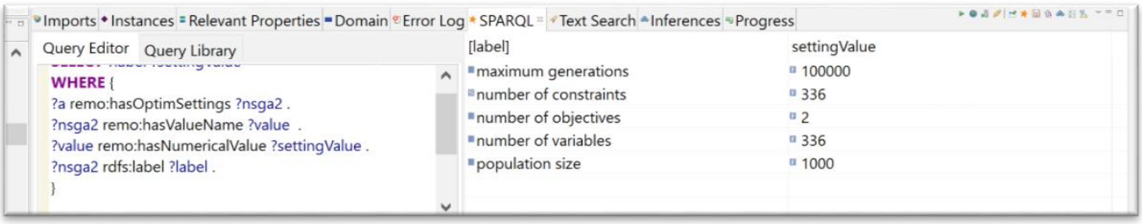



**Figure 96. SPARQL query to retrieve optimisation settings needed for district optimisation**

- The initial values for all the decision variables are also taken into account in the optimisation model. This can be retrieved using the following query:

```
SELECT ?label ?readLocationID
WHERE {
?a remo:hasDecisionVariable ?dv .
?a a remo:UseCases_District_Optimisation_TypicalDemand .
?dv remo:has_locationEMS ?c .
?dv a remo:ProductionScheduleHeat_Optimised .
?dv rdfs:label ?label .
?c remo:hasLocationString_Read ?readLocationID.
}
```

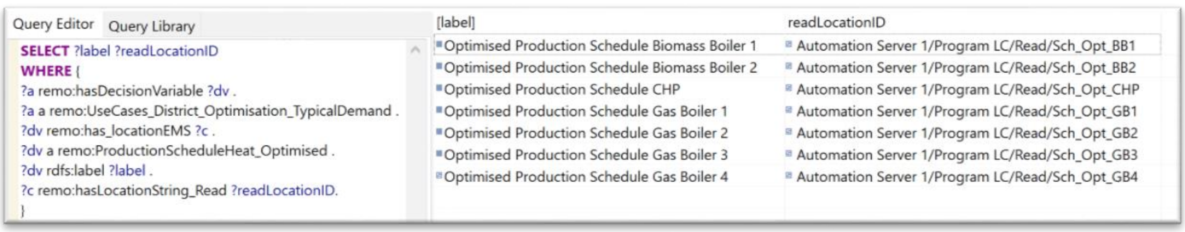



**Figure 97. SPARQL query to read initial values of decision variables.**

- For optimisation to take place other parameters are also needed (refer to Section 4.2.2) which can be retrieved using other SPARQL queries. In this use case, for example, the typical demand schedules are also required for each consumer building in the district because the analytical model requires these for calculations. The query for this is shown below:

```
SELECT ?buildingName ?typicalHeatDemandReadLocation
WHERE {
 ?x a remo:EnergyConsumerBuilding .
?x rdfs:label ?buildingName .
?x remo:hasHeatDemand ?y .
?y remo:has_locationBMS ?z .
?z remo:hasLocationString_Read ?typicalHeatDemandReadLocation .
              }
```

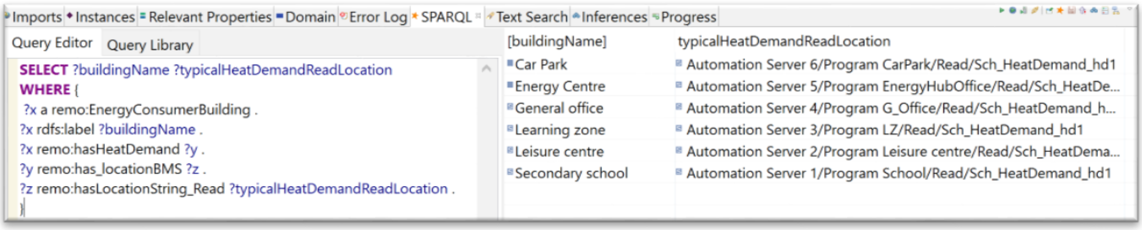



**Figure 98. SPARQL query to read typical demand profiles of each building**

- Once the optimisation is complete, the optimised production schedules for each energy source in Ebbw Vale are set after retrieving their setpoint location (as they are decision variables of the use case). The setpoint location of these individuals and their labels can be retrieved using the query, as shown below:

```
SELECT ?label ?setLocationID
WHERE {
?a remo:hasDecisionVariable ?dv .
?a a remo:UseCases_District_Optimisation_TypicalDemand .
?dv remo:has_locationEMS ?c .
?dv a remo:ProductionScheduleHeat_Optimised .
?dv rdfs:label ?label .
?c remo:hasLocationString_Write ?setLocationID.
}
```

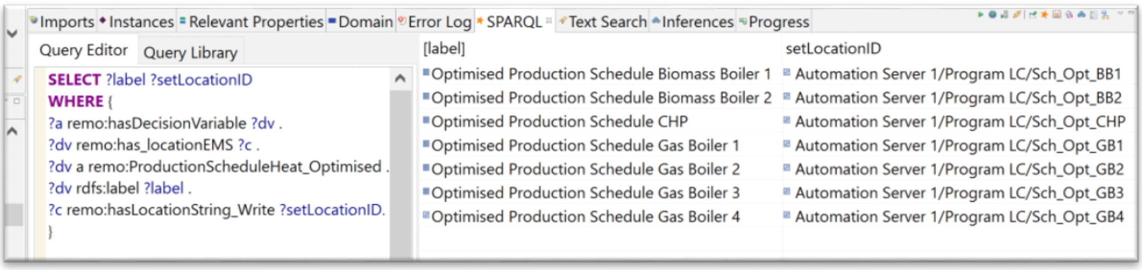



**Figure 99. SPARQL query to retrieve setpoint location of decision variables of use case 3**

- The objective values of the optimised solution can also be set in the EMS once its location has been identified using the SPARQL query shown below.

```
SELECT ?label ?objValue
WHERE {
?a remo:hasOptimObjective ?dv .
?a a remo:UseCases_District_Optimisation_TypicalDemand .
?dv remo:has_locationEMS ?location .
?dv rdfs:label ?label .
?location remo:hasLocationString_Write ?objValue.
}
```

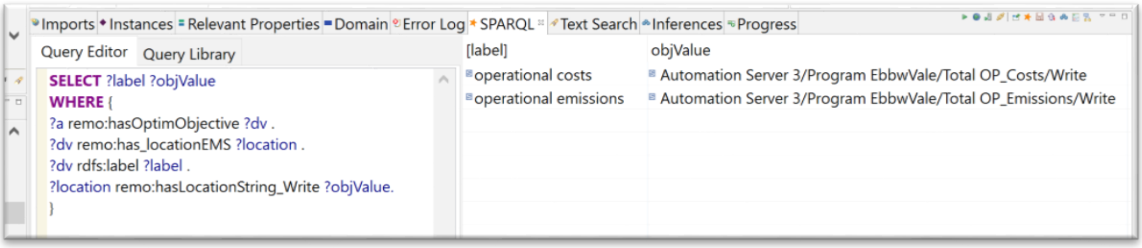



**Figure 100. SPARQL query to retrieve setpoint location of optimisation objectives of use case 3**

- The overall workflow of this use case is shown below in Figure 101.

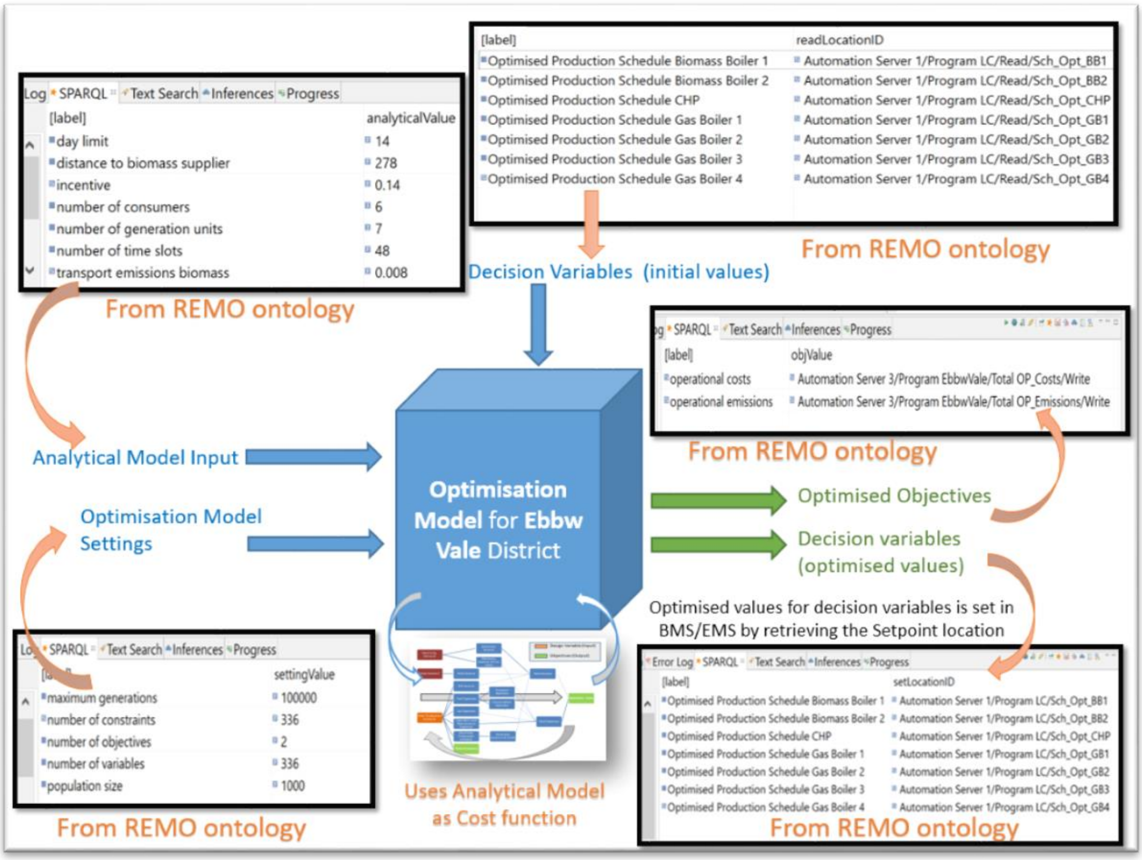



**Figure 101. Results of SPARQL queries being used for the optimisation model of use case 3.**

**Sample use case 4 - Training of heat demand prediction model for Leisure Centre building**

- Historical data and 'labels' can be retrieved for each individual input and output parameter of the ANN model of this use case using a SPARQL query. Endpoint location of ANN input data can be queried with the following query:

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnInput ?b .
?a a remo:UseCases_Building_Prediction_OverallDemandProfile_Heat .
?b remo:has_locationBMS ?c .
?c remo:hasHistoricalDataLocationString ?locationID.
                                        ?b rdfs:label ?label }
```

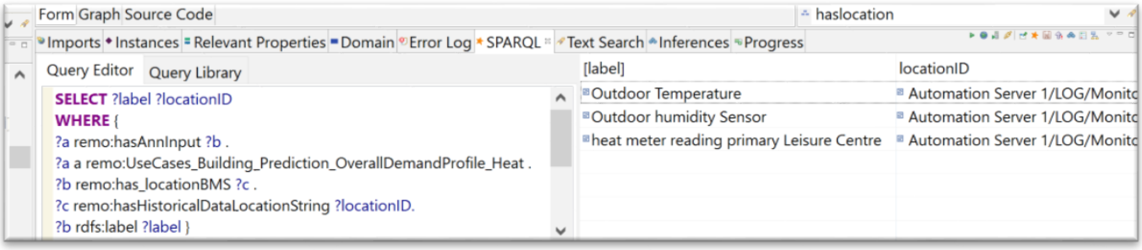



**Figure 102. SPARQL query to retrieve ANN input data of use case 4 and its results**

Similarly, endpoint location for ANN output data can be queried with the following query:

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnOutput ?b .
?a a remo:UseCases_Building_Prediction_OverallDemandProfile_Heat .
?b remo:has_locationBMS ?c .
?c remo:hasHistoricalDataLocationString ?locationID.
            ?b rdfs:label ?label }
```

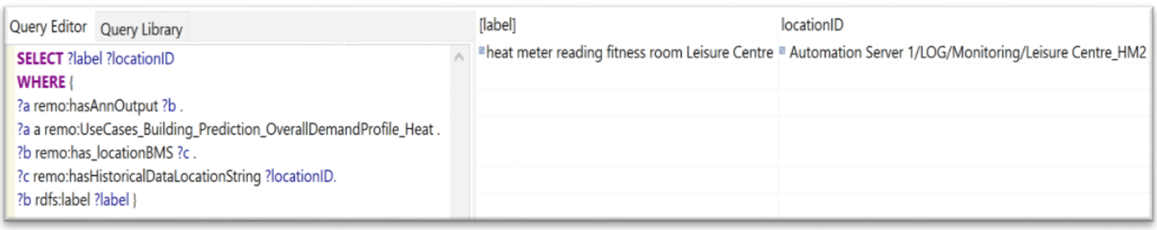



**Figure 103. SPARQL query to retrieve ANN output data of use case 4 and its results.**

- This data is then post-processed and used to train the ANN model for the Leisure Centre building as shown below in Figure 104. Once trained, the ANN model is stored as an executable file as followed in the SportE2 project. The executable file can then be used for prediction of overall heat demand for this building. Similarly, every building in the district will have its own independent ANN model.

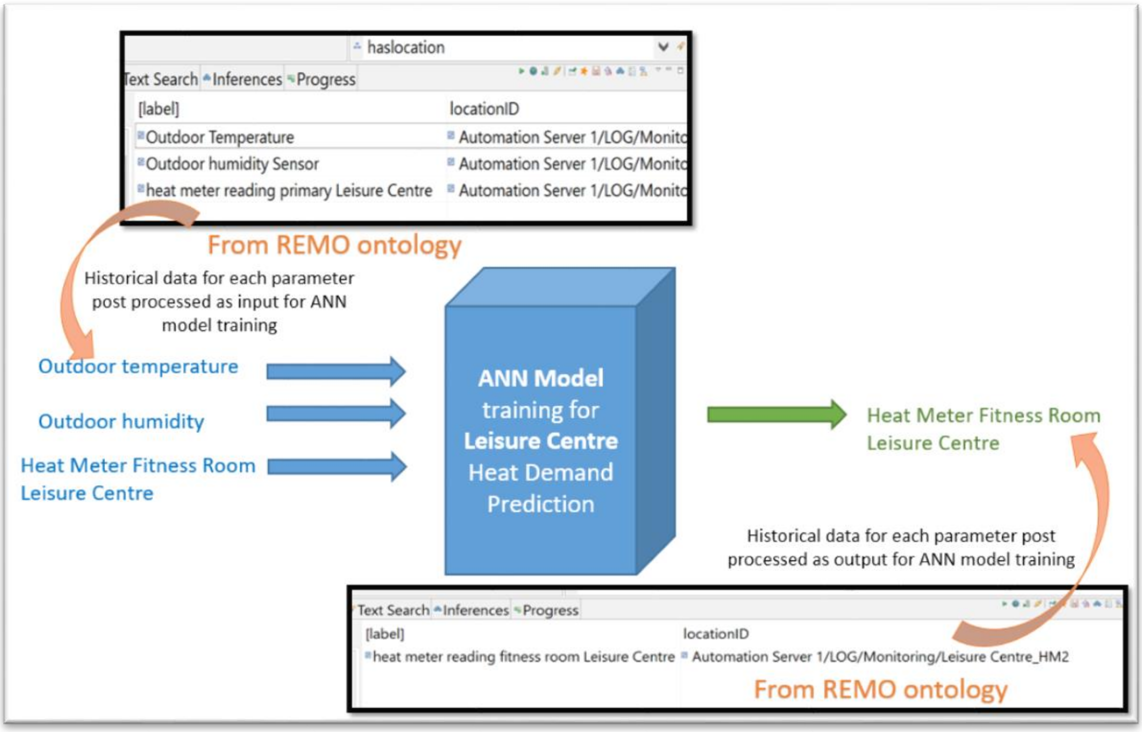



**Figure 104. Results of SPARQL queries being used to train ANN model of use case 4.**

**Sample use case 5 - Running of the heat demand prediction model for the Leisure Centre building**

- Real-time data and 'labels' can be retrieved for each ANN input of the model by using a SPARQL query. ANN input can be queried using the following queries:

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnInput ?b .
?a a
remo:UseCases_Building_Prediction_Model_OverallDemandProfile_Heat
?b a remo: ForecastedWeatherSchedule_Temperature
?b remo:has_locationEMS ?c .
?c remo:hasLocationString_Read ?locationID.
        ?b rdfs:label ?label }
```

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnInput ?b .
?a a
remo:UseCases_Building_Prediction_Model_OverallDemandProfile_Heat
.
?b a remo: ForecastedWeatherSchedule_Humidity
?b remo:has_locationEMS ?c .
?c remo:hasLocationString_Read ?locationID.
            ?b rdfs:label ?label }
```

```
SELECT ?label ?locationID
WHERE {
?a remo:hasAnnInput ?b .
?a a
remo:UseCases_Building_Prediction_Model_OverallDemandProfile_Heat
.
?b a remo:MeterReadings_Primary_heat
?b remo:has_locationBMS ?c .
?c remo:hasLocationString_Read ?locationID.
            ?b rdfs:label ?label }
```

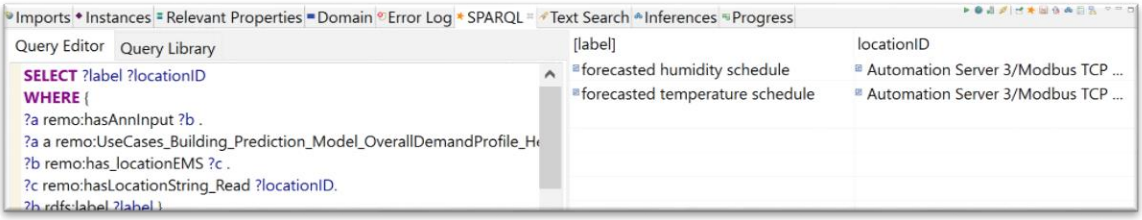



**Figure 105. SPARQL query to retrieve endpoint location of ANN input (weather parameters) of use case 5 and its results**

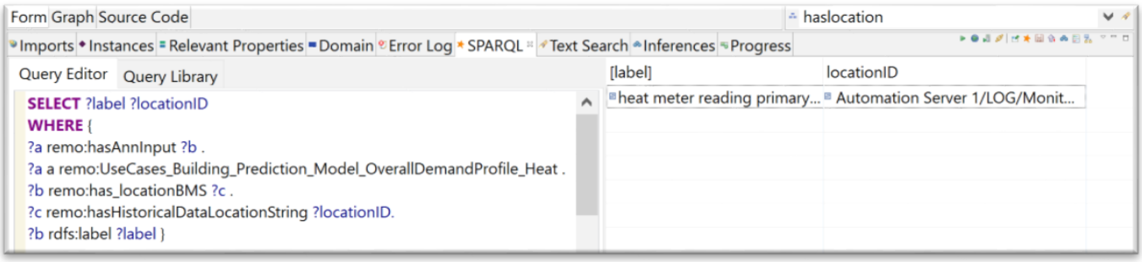



**Figure 106. SPARQL query to retrieve endpoint location of ANN input (meter data) of use case 5 and its results**

- Post-processing of the retrieved data from the endpoint location above is needed, and the following information is used as ANN input:
  - From the historical meter readings of the primary heat meter for the Leisure Centre, the previous 24-hour demand is retrieved.
  - Along with this the forecasted 24-hour weather profile is chosen, which includes the outdoor temperature and outdoor humidity.

- This data is then used as input to the pre-trained ANN model of the Leisure Centre building from sample use case 4 as shown in Figure 108.
- The output of the ANN model of the Leisure centre building is set under the individuals of the `DemandSchedule_Heat_Predicted` class. The schedule can be set in the BMS. This setpoint location can be queried using the following SPARQL query:

```
SELECT ?label ?setLocationID
WHERE {
?a remo:hasAnnOutput ?b .
?a a remo:UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .
?b remo:has_locationBMS ?c .
?c remo:hasLocationString_Write ?setLocationID.
        ?b rdfs:label ?label }
```

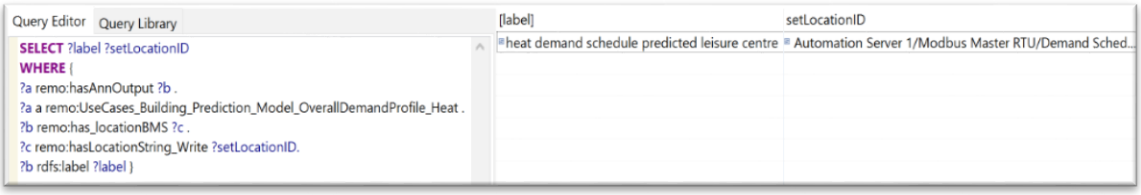



**Figure 107. SPARQL query to retrieve setpoint location of ANN output of use case 5 and its results**

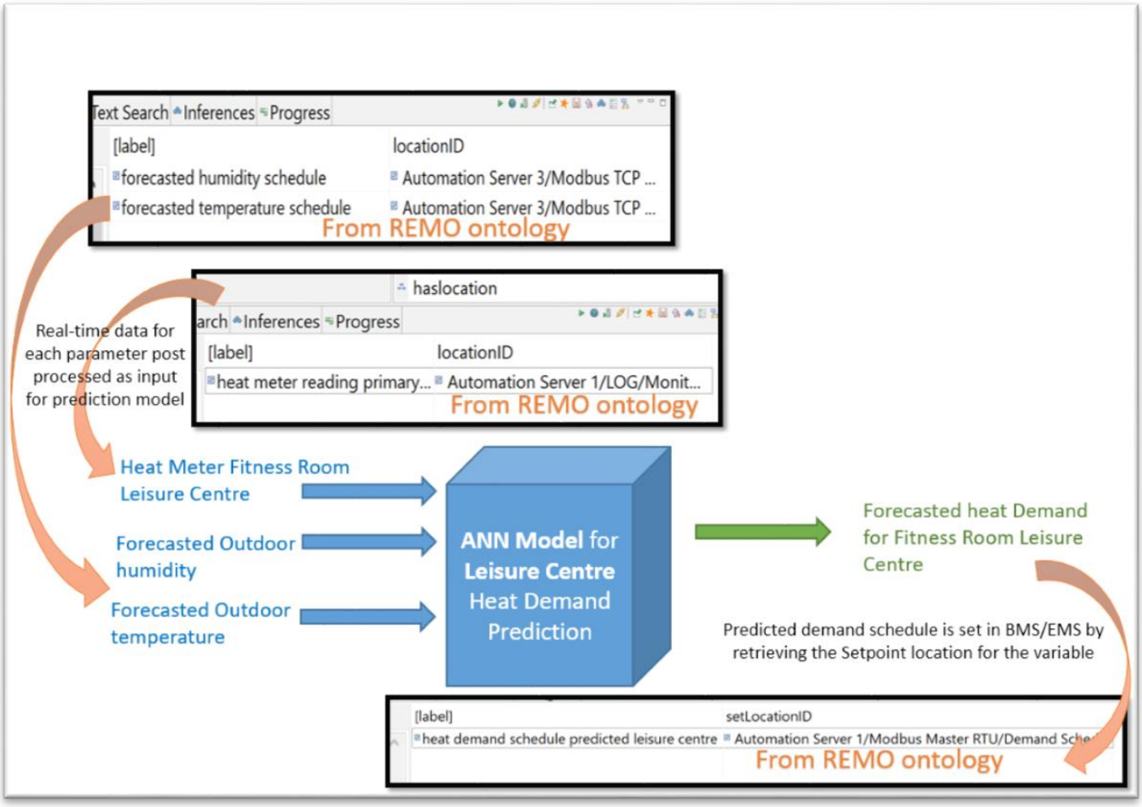



**Figure 108. Results of SPARQL queries used to run the ANN model of use case 5.**

**Sample use case 6 - District optimisation of Ebbw Vale using predicted demand profiles of buildings**

This use case is very similar to use case 3.

- SPARQL queries help retrieve static parameters and their labels using the same query as covered in use case 3.
- A set of SPARQL queries is used to retrieve some of the other relevant parameters needed for the district optimisation and analytical model to run. The queries are similar to use case 3 again, but the only difference here is that predicted demand is used from the class `DemandSchedule_Heat_Predicted` instead of typical demand profiles. This can be retrieved using the following query:

```
SELECT ?buildingName ?predictedHeatDemandReadLocation
WHERE {
 ?x a remo:EnergyConsumerBuilding .
?x rdfs:label ?buildingName .
?x remo:hasPredictedHeatDemand ?y .
?y remo:has_locationBMS ?z .
?z remo:hasLocationString_Read ?typicalHeatDemandReadLocation .}
```

- Once the optimisation is complete, these production schedules are set under the individuals of the class `ProductionScheduleHeat_Optimised`, as followed in use case 3. This can be done by retrieving the location of this schedule from the EMS.
- The objective values of the optimised solution can also be set in the EMS once its location has been identified. The location is retrieved by using queries similar to use case 3.

## 7.2. Replication of framework to other sites

Figure 109 below shows the general methodology that needs to be adopted in the framework for any future district site. Using ontologies allows the framework to be easily reusable for any other site.

For any new site, the ontology is first instantiated as explained in Section 6.1.3. Importantly, the instantiated ontology should contain:

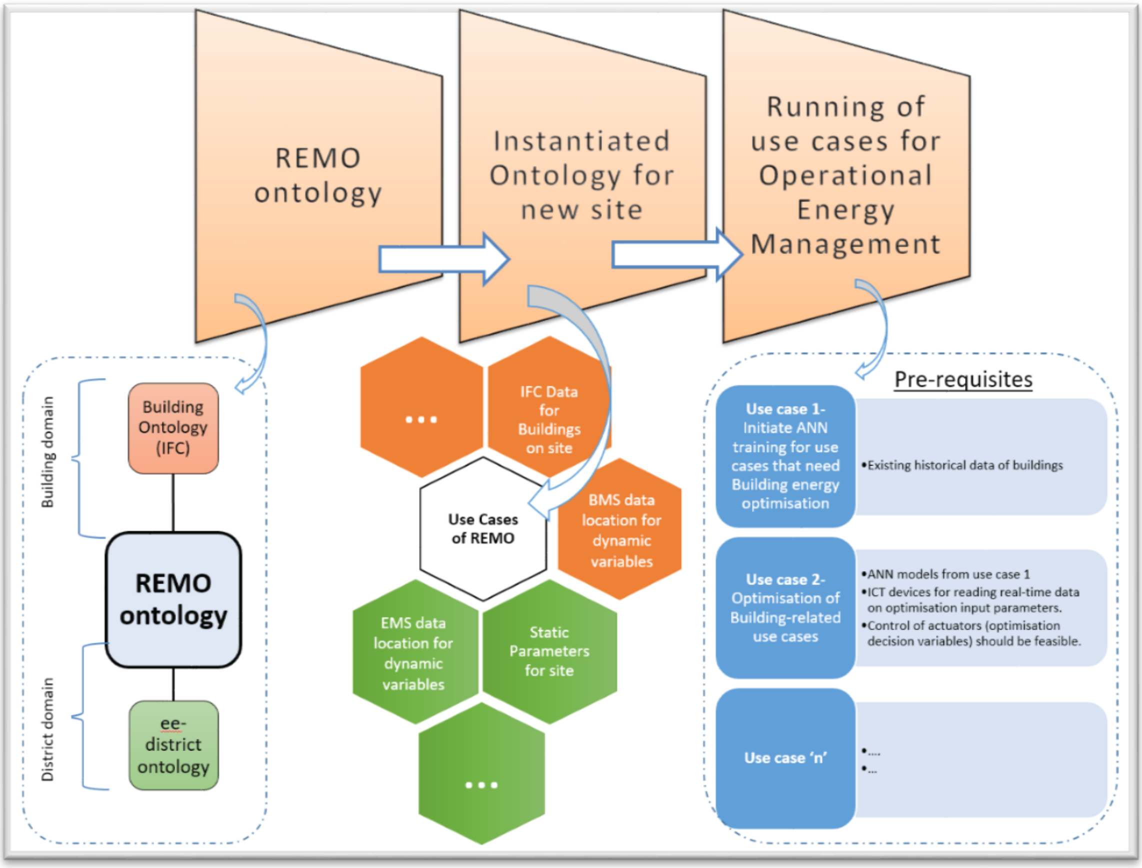



**Figure 109. Framework replication plan**

- The new static parameters of the site and their numerical values.
- Locations of dynamic parameters (in BMS/EMS), which would enable the ontology to point out the location of data for the functional layer of the framework to run.
- Use cases need to be instantiated as well.

Post-reasoning, the ontology can then be queried for information needed to run all the use cases as covered in Section 7.1.3. These use cases do have pre-requisites, and those need to be met. Use cases 1 and 4 are run during the set-up phase of the framework for a new district for the training of the ANN models. Once this is complete, use cases 2, 3, 5, and 6 (from Section 5.3) are available to be queried for real-time energy management. Some of the use cases require analytical and optimisation models, which can be reused across various sites as long as they are utilised for the pre-defined use cases in the ontology. Some of these use cases also require ANN models. The ANN models applied on one site and its buildings cannot be reused for a new site, and hence they would have to be pre-trained with data. Data, therefore, is quite critical to get the framework to run. In future, **REMO** ontology can be further extended to incorporate more use cases related to district/building optimisation scenarios.

# 8. Conclusions and future work

## 8.1. Conclusions

The energy management solution today should be able to make smarter decisions considering the multi-objectives involved in both supply and demand side domains, working alongside with automation systems that provide real-time actuation. The author's involvement in EU projects through action research lays the foundation to this research. Through this action research, the author affirmed that real-time energy management is possible today using artificial intelligence solutions. Action research concluded that prediction models using ANN and, multi-objective optimisation using genetic algorithms are highly effective to solve the complex multi-objective problems at both the demand- and supply-side level in real-time. The demand-side optimisation approach was tested and validated through the SportE2 project and on an average brings about 36% savings in energy. The district energy supply side optimisation, was also promising and resulted in 31.8% increase in profits and 36% emissions savings.

However, one of the biggest problems identified through the literature reviewed and also affirmed through action research is the growing gap between demand- and supply-side energy management failing to take a holistic approach. The author addressed this issue by proposing to use the reduced demand profiles from building demand-side optimisation and consequently using these for optimisation of the supply-side in the district. **REMO** ontology, developed here, proposes to facilitate seamlessly this unique method of harmonisation of demand- and supply-side energy management. Another problem identified was the silo-oriented approach to decision-making in each of this domain. However, **REMO** ontology, built as a cross-domain knowledge-base is able to consider not only the multi-scale nature of the problem, i.e. considering both demand-side and supply-side optimisation, but also take into account the multi-objectives (costs, emissions, and efficiency) involved in an optimisation problem.

Although, the research only stresses on **REMO** ontology development and its validation, the work showcases the potential application of **REMO** in supporting a real-time energy management framework, through the various use cases presented. Here, the ontology acts as the brain of the framework linking heterogeneous technologies, systems, and information sources. The validation chapter demonstrates how **REMO** ontology can be queried to run these use cases and how the ontology brings together various data domains and technologies. Moreover, the framework can also be easily replicated and used for new sites, provided the ontology is re-instantiated. Reusability is one of the advantages of using ontologies in the framework, and it is made possible through rule axioms features in the domain model. These rule axioms capture the intelligence behind the AI and numerical models in the domain.

One of the biggest challenges of implementing this framework would be that it requires a lot of data. Data (either historical or simulation) is necessary for the use cases of the overall framework to run especially because both the supply- and demand-side optimisation depends on ANN models. For example, for demand-side, the artificial intelligence-based optimisation requires adequate data for pre-training of the ANN models which are later used as a cost function for the optimisation problem. Whereas, in the case of district optimisation, the ANN models are needed for day-ahead demand forecast for each building, which is input to the district analytical model.

## 8.2. Contributions

The first contribution by the author in this research is the district analytical model developed for real-time optimisation of the supply-side of the district. This model considered many data domains and was used to compute both operational emissions and costs in the district. Multi-objective optimisation using the NSGA-II algorithm in this analytical model helped optimise the production schedules of heat energy sources, optimising the costs and emissions. Seldom before have all the different domains been considered for operational optimisation. Moreover, the author demonstrates how this optimisation model can work along with the real-time demand optimisation methodology developed in the SportE2 project. This harmonisation between demand and supply side optimisation is a valid contribution as it is considered a big gap in today's district energy management solutions.

**REMO** ontology, which was built to facilitate this harmonisation working with automation systems and AI solutions, can be considered as the primary contribution of this research. Using ontologies in this research brought more than interoperability benefits. Here, the ontology not only facilitate requirements of the optimisation and prediction models needed for the demand and supply side energy management to work, but also captured knowledge behind these models. This knowledge comes from the experience gained by the author through action research and was modelled into the domain ontology through rule axioms. This methodology is a unique contribution to the field of knowledge. Which also meant, this knowledge can be replicated for future sites. Using ontologies in such a manner has not been attempted before.

The research also highlights how IFC data model that is used in BIM today, is not enough for a holistic energy analysis as there were very few overlaps between IFC concepts and REMO taxonomy. However, frameworks such as the one proposed in this research can be the future for BIM based holistic energy analysis, because REMO ontology here complements BIM models linking all the knowledge needed for real-time energy management with IFC models.

The author has also made contributions to the following research papers:

**Journal Papers**

- Jayan, B., Li, H., Rezgui, Y., Hippolyte, J.-L. and Howell, S. 2016. An Analytical Optimization Model for Holistic Multiobjective District Energy Management - A Case Study Approach. *International Journal of Modeling and Optimization* 6(3), pp. 156–165.

- Petri, I., Li, H., Rezgui, Y., Chunfeng, Y., Yuce, B. and Jayan, B. 2014a. A HPC based cloud model for real-time energy optimisation. *Enterprise Information Systems*, pp. 1–21.

- Petri, I., Li, H., Rezgui, Y., Chunfeng, Y., Yuce, B. and Jayan, B. 2014b. A modular optimisation model for reducing energy consumption in large scale building facilities. *Renewable and Sustainable Energy Reviews*, pp. 990–1002.

- Yang, C., Li, H., Rezgui, Y., Petri, I., Yuce, B., Chen, B. and Jayan, B. 2014. High throughput computing based distributed genetic algorithm for building energy consumption optimization. *Energy and Buildings* 76, pp. 92–101.

- Yuce, B., Li, H., Rezgui, Y., Petri, I., Jayan, B. and Yang, C. 2014. Utilizing artificial neural network to predict energy consumption and thermal comfort level: An indoor swimming pool case study. *Energy and Buildings* 80, pp. 45–56.

**Conference Papers**

- Jayan, B., Li, H., Rezgui, Y., Hippolyte, J.L., Yuce, B., Yang, C. and Petri, I. 2014. An ontological approach to intelligent energy management in building. In: *EG-ICE 2014, European Group for Intelligent Computing in Engineering - 21st International Workshop: Intelligent Computing in Engineering 2014*. Cardiff University.

- Hippolyte, J.-L., Rezgui, Y., Haijiang, L. and Jayan, B. 2014. An ee-district ontology to support the development of the ee-District Information Model of the RESILIENT

project. In: *EEBuilding Data Models : Energy Efficiency Vocabularies & Ontologies*. Nice: EEB Data Models Community, pp. 106–119.

- Howell, S., Hippolyte, J.-L., Jayan, B., Reynolds, J. and Rezgui, Y. 2016. Web-based 3D Urban Decision Support through Intelligent and Interoperable Services. Proceedings of the 2nd IEEE International Smart Cities Conference. Trento; Italy, 12 September, 2016.

## 8.3. Future work

This research details the semantic layer and functional layer of the framework that can be used for real-time energy management and optimisation. In this research, only the semantic layer was completed and tested, as shown below in Figure 110 in blue chevron.

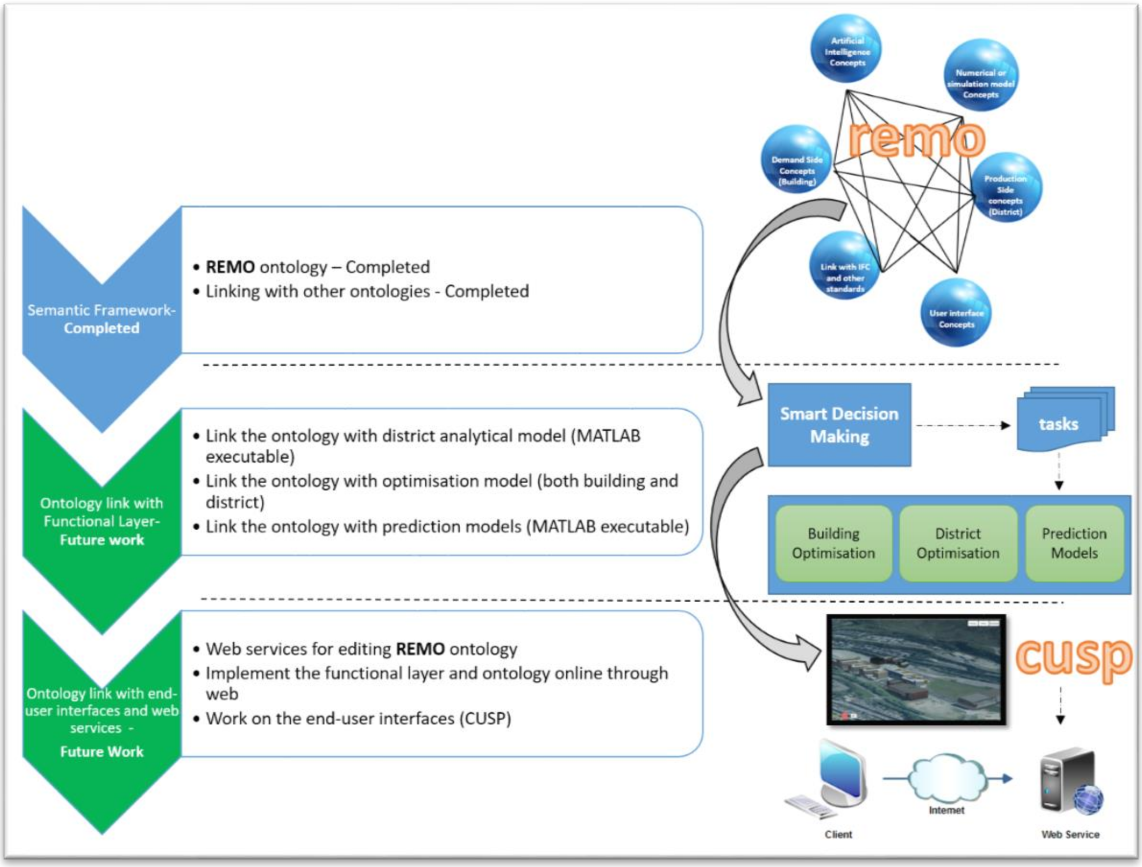



**Figure 110. Future work**

The optimisation and prediction solutions were also tested and validated individually. Future work should look into two main steps as shown in the figure above in green chevrons:

1. *Developing the link between the semantic layer and the functional layer*

Here, the communication between the layers of the overall framework needs further development. Future work should look into integrating these two layers and demonstrate their working with each other. To achieve this, in reality, script files are needed, as shown below in Figure 111. These script files are computer codes and can be used for many purposes. For example, these can be written to execute a set of SPARQL queries based on the tasks the user wants to run. Here, the SPARQL engine can also be used to query the ontology, similar to the approach in the Resilient project (Section 4.2.1). Script files can also be used to collect responses from the ontology and generate input files for prediction or optimisation models, consequently triggering these models.

2. *Linking the semantic layer with end-user interfaces and implementing it in web services*

Ontologies can be very dynamic in nature, with information needing to be changed now and then, and therefore they should be able to be easily edited. Therefore, web implementation of **REMO** ontology is necessary to provide a user-friendly approach to modifying and instantiating the ontology. For example, in the Resilient project, the web implementation of the ontology allowed easy interaction with the other modules such as simulation models and multi-agent systems (which were also implemented online). Similarly, **REMO** ontology, which is a high-level semantic representation of the district, would need to provide knowledge to different energy management software. For example, the ontology has key entities required for running energy optimisation functionalities, which can be useful for third-party applications. Here, encapsulating the ontology into a web service makes it easier for applications to access the knowledge from it. These web services can also easily be linked to the end-user interface such as CUSP, which was discussed under Section 6.3 in Chapter 6.

Overall working of the framework in the future could be as shown in Figure 111 below:

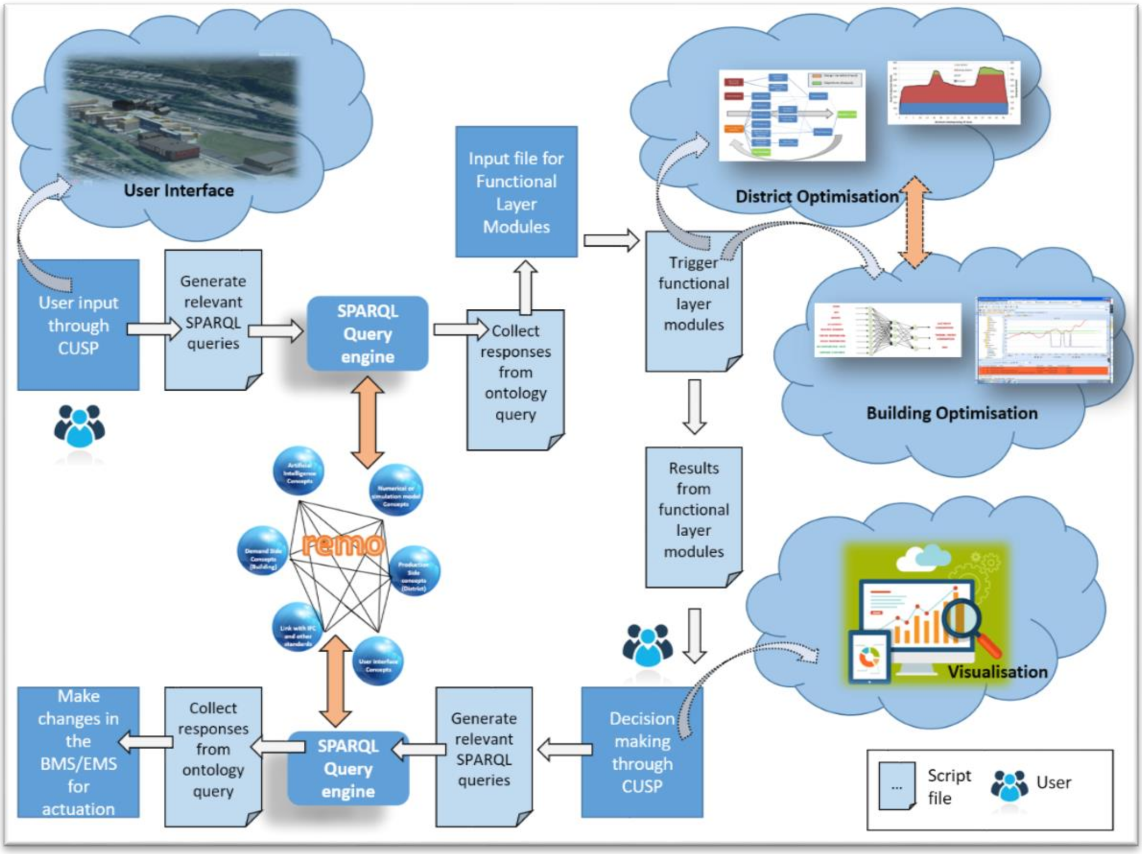



**Figure 111. Detailed framework and its workflow for future development**

The communication between these layers as shown through the script file needs to be completed in the future for the overall testing of the framework.

Some of the other future work should include looking into:

1. *Further investigation on* **ifcOWL** *linking with* **REMO**

   IFC ontology has been mapped with **REMO** ontology, even though there are very few classes and properties with similar semantic meaning in both ontologies. Future work should look into instantiating **REMO** ontology through an instantiated **ifcOWL** ontology, which would make the instantiation process easier for the user provided IFC models for the buildings are available. This research does not study this in detail because **ifcOWL** is still in the process of being standardised.

2. *Further investigation in the functional layer*

   - The district optimisation methodology developed as a part of the holistic energy management methodology needs to be tested with different multi-objective optimisation algorithms to look for better results. In this research, with the ontology development being the priority, less time was spent on trying to improve the multi-objective optimisation of the supply side. The analytical model can also be further developed by linking this to dynamic simulation models to take into account latency effect, time constants and heat losses. In this research, the case study does not include any renewable technologies, but these technologies might have to be adopted into the generic analytical model in the future as renewables will be very visible in the generation mix for the future sites, following a trend in the energy markets. The optimisation models and the analytical model currently are very specific to the Ebbw Vale site; they need to be made generic to be able to work with the other sites.

- Only preliminary work was conducted on the day-ahead demand forecasts for the buildings as shown in the conclusions of action research Section 4.3. Further research/testing on these models is needed. These prediction models of each building can also be integrated with the district analytical model and run together if needed.

The features of the ontology and its working are validated. The validation of the numerical and optimisation models was mainly through the action research. In the future, a real pilot site needs to be tested with both the supply and demand side optimisation facilitated through the ontology and the overall framework proposed here.

# Bibliography

Ahmadi-Karvigh, S., Becerik-Gerber, B. and Soibelman, L. 2016. A framework for allocating personalized appliance-level disaggregated electricity consumption to daily activities. *Energy and Buildings* 111, pp. 337–350.

Allegrini, J., Orehounig, K., Mavromatidis, G., Ruesch, F., Dorer, V. and Evins, R. 2015. A review of modelling approaches and tools for the simulation of district-scale energy systems. *Renewable and Sustainable Energy Reviews* 52, pp. 1391–1404.

Amires and Bumblebeestudio.eu. 2016. *Ambassador Project -Mission and Objectives* [Online]. Available at: http://ambassador-fp7.eu/mission-objectives/ [Accessed: 10 January 2016].

Anzaldi, G., Corchero, A., Gutu, D., Marinov, M., Dibley, M.J. and McGlinn, K. 2012. *Intelligent Building Energy Management through Holistic Knowledge Based Approach*. IOS Press.

d'Aquin, M., Zimmermann, A., Dzbor, M., Iglesias, M., Jacques, Y., Caracciolo, C., Aranda, C.B., et al. 2008. D1. 1.3 NeOn Formalisms for Modularization: Syntax, Semantics, Algebra. [Online]. Available at: http://www.neon-project.org/deliverables/WP1/NeOn_2008_D1.1.3.pdf [Accessed: 26 March 2016].

Arnette, A. and Zobel, C.W. 2012. An optimization model for regional renewable energy development. *Renewable and Sustainable Energy Reviews* 16(7): pp. 4606–4615.

ASCD (2016). *What Is Action Research?* [Online]. Available at: http://www.ascd.org/publications/books/100047/chapters/What-Is-Action-Research%C2%A2.aspx [Accessed: 2 July 2016].

Aung, H.N., Khambadkone, A.M., Srinivasan, D. and Logenthiran, T. 2010. Agent-based intelligent control for real-time operation of a microgrid. In: *2010 Joint International Conference on Power Electronics, Drives and Energy Systems, PEDES 2010 and 2010 Power India*.

Ayata, T., Arcaklıoğlu, E. and Yıldız, O. (2007). Application of ANN to explore the potential use of natural ventilation in buildings in Turkey. *Applied Thermal Engineering* 27(1), pp. 12–20.

Baños, R., Manzano-Agugliaro, F., Montoya, F.G., Gil, C., Alcayde, A. and Gómez, J. 2011. Optimization methods applied to renewable and sustainable energy: A review. *Renewable and Sustainable Energy Reviews* 15(4), pp. 1753–1766.

Baumgärtel, K., Katranuschkov, P. and Scherer, R.J. 2015. Ontology-controlled Energy Simulation Workflow. In: *Sustainable Places 2015*. Italy: Sigma Orionis.

Bazjanac, V. 2008. IFC BIM-based methodology for semi-automated building energy performance simulation. *Lawrence Berkeley National Laboratory* [Online]. Available at: http://escholarship.org/uc/item/0m8238pj.pdf [Accessed: 30 March 2016].

Bazmi, A.A. and Zahedi, G. 2011. Sustainable energy systems: Role of optimization modeling techniques in power generation and supply: A review. *Renewable and Sustainable Energy Reviews* 15(8), pp. 3480 – 3500.

Ben-Nakhi, A.E. and Mahmoud, M.A. 2011. Case Study of Applying Different Energy Use Modeling Methods To An Existing Building.

Ben-Nakhi, A.E. and Mahmoud, M.A. 2004. Cooling load prediction for buildings using general regression neural networks. *Energy Conversion and Management* 45(13-14), pp. 2127–2141.

Bichiou, Y. and Krarti, M. 2011. Optimization of envelope and HVAC systems selection for residential buildings. *Energy and Buildings* 43(12), pp. 3373–3382.

Biomass Energy Centre 2011. *Estimated Carbon Dioxide Emissions for Freight Transport*. Crown.

Biomass Energy Centre 2011b. *Renewable Heat Incentive*. Biomass Energy Centre.

Böhms, H.M., Plokker, W., Charvier, B., Madrazo, L. and Sicilia, A. 2010. IntUBE energy information integration platform. In: *eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the European Conference on Product and Process Modelling 2010*. pp. 339–344.

Bonino, D., Corno, F. and De Russis, L. 2015. *Poweront: An Ontology-Based Approach for Power Consumption Estimation in Smart Homes*. Springer Verlag.

Brank, J., Grobelnik, M. and Mladenic, D. 2005. A survey of ontology evaluation techniques. In: *Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2005)*. pp. 166–170.

Brazier, F., Poutre, H.L., Abhyankar, A.R., Saxena, K., Singh, S.N. and Tomar, K.K. 2015. A review of multi agent based decentralised energy management issues. In: IEEE, pp. 1–5.

British Standards 2011. *Energy Management System Application Program Interface (EMS-API) - Part 301: Common Information Model (CIM) Base (IEC 61970-301:2011)*. British Standards Institution.

British Petroleum 2013. *Sustainability Review 2012*. British Petroleum.

Brownlee, J. 2015. *Genetic Algorithm - Clever Algorithms: Nature-Inspired Programming Recipes* [Online]. Available at: http://www.cleveralgorithms.com/nature-inspired/evolution/genetic_algorithm.html [Accessed: 21 August 2016].

Brundu, F.G., Patti, E., Giudice, M.D., Osello, A., Macii, E. and Aquaviva, A. 2015. DIMCloud: A Distributed Framework for District Energy Simulation and Management. In: Giaffreda, R., Vieriu, R.-L., Pasher, E., Bendersky, G., Jara, A. J., Rodrigues, J. J. P. C., Dekel, E., et al. eds. *Internet of Things. User-Centric IoT*. Cham: Springer International Publishing, pp. 331–338.

buildingSMART International Ltd 2016. *buildingSMART- IFC Overview* [Online]. Available at: http://www.buildingsmart-tech.org/specifications/ifc-overview [Accessed: 10 January 2016].

Buoro, D., Pinamonti, P. and Reini, M. 2014. Optimization of a Distributed Cogeneration System with solar district heating. *Applied Energy* 124, pp. 298–308.

Caffarel, J., Jie, S., Olloqui, J. and Martínez, R. 2012. Bat-MP: An Ontology-Based Energy Management Platform. In: Bravo, J., López-de-Ipiña, D. and Moya, F. (eds.) *Ubiquitous Computing and Ambient Intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 9–16.

Caffarel, J., Jie, S., Olloqui, J., Martínez, R. and Santamaría, A. 2013. Implementation of a Building Automation System based on semantic modeling. *Journal of Universal Computer Science* 19(17), pp. 2543–2558.

Cai, Y.P., Huang, G.H., Lin, Q.G., Nie, X.H. and Tan, Q. 2009. An optimization-model-based interactive decision support system for regional energy management systems planning under uncertainty. *Expert Systems with Applications* 36(2), pp. 3470 – 3482.

Camacho, R., Carreira, P., Lynce, I. and Resendes, S. 2014. An ontology-based approach to conflict resolution in Home and Building Automation Systems. *Expert Systems with Applications* 41(14), pp. 6161–6173.

Cardiff University (2014). *Deliverable 2.4 – Updated District Information Model* [Online], Available at: http://www.resilient-project.eu/ [Accessed: 20 January 2015].

Cardiff University (2013). *Deliverable 4.3 - Why Module as a Product* [Online], Available at: http://www.sporte2.eu/public-documents [Accessed: 05 January 2014].

Cardiff University (2015). *Deliverable 4.4 - Why Module Complete* [Online], Available at: http://www.sporte2.eu/public-documents [Accessed: 10 January 2016].

Carlos Gershenson 2003. *Artificial Neural Networks for Beginners*. Sussex: University of Sussex.

Cherifi, H. (ed.) 2011. *Digital Information and Communication Technology and Its Applications: International Conference, DICTAP 2011, Dijon, France, June 21-23, 2011; Proceedings. Pt. 2: [...]*. Heidelberg: Springer.

Chicco, G. and Mancarella, P. 2008. Assessment of the greenhouse gas emissions from cogeneration and trigeneration systems. Part I: Models and indicators. *Energy 33(3)* , pp. 410–417.

Commissariat à l'Energie Atomique et aux Energies Alternatives 2015. *D2.7 - Final ICT Framework for Real-Time Energy Management at District-Level*. FRANCE: Commissariat à l'Energie Atomique et aux Energies Alternatives.

Committee on Climate Change 2013. *Fourth Carbon Budget Review*. London: Committee on Climate Change.

Corrado, V., Ballarini, I., Madrazo, L. and Nemirovskij, G. 2015. Data structuring for the ontological modelling of urban energy systems: The experience of the SEMANCO project. *Sustainable Cities and Society* 14(1), pp.223–235.

Corry, E., Pauwels, P., Hu, S., Keane, M. and O'Donnell, J. 2015. A performance assessment ontology for the environmental and energy management of buildings. *Automation in Construction* 57, pp. 249–259.

Costa, G. and Madrazo, L. 2015. Connecting building component catalogues with BIM models using semantic technologies: an application for precast concrete components. *Automation in Construction* 57, pp. 239–248.

Cricchio, F., Mastrodonato, C. and Reccardo, D. 2012. *D2.2.1: Report on State-of-the-Art (International) ICT-Based Innovation Projects*. IREEN Consortium.

Culatta, R. 2015. *Action Research* [Online]. Available at: http://www.innovativelearning.com/teaching/action_research.html [Accessed: 2 July 2016].

Dam, K.H. van 2009. *Capturing Socio-Technical Systems with Agent-Based Modelling*. PhD Thesis, Delft University of Technology.

Daniele, L., den Hartog, F. and Roes, J. 2015. *Created in Close Interaction with the Industry: The Smart Appliances REFerence (SAREF) Ontology*. Springer Verlag.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. 2002. A fast and elitist multi-objective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on* 6(2), pp. 182–197.

Den Hartog, F., Daniele, L. and Roes, J. 2015. Toward semantic interoperability of energy using and producing appliances in residential environments. In: *2015 12th Annual IEEE Consumer Communications and Networking Conference*, *CCNC 2015*. Institute of Electrical and Electronics Engineers Inc., pp. 170–175.

Vrandecic, D. 2010. *Ontology Evaluation*. PhD Thesis, Karlsruhe Institute for Technology.

UK Government Department of Energy and Climate Change 2008. *CHP Finance*. Department of Energy and Climate Change. Available at: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/345190/Part_5_CHP__Finance.pdf [Accessed: 3 January 2016].

UK Government Department of Environment, Food & Rural Affairs 2014. *Greenhouse Gas Conversion Factor Repository*. Available at: http://www.ukconversionfactorscarbonsmart.co.uk/ [Accessed: 3 January 2016].

Diakaki, C., Grigoroudis, E., Kabelis, N., Kolokotsa, D., Kalaitzakis, K. and Stavrakakis, G. 2010. A multi-objective decision model for the improvement of energy efficiency in buildings. *Energy* 35(12), pp. 5483–5496.

Diakaki, C., Grigoroudis, E. and Kolokotsa, D. 2008. Towards a multi-objective optimization approach for improving energy efficiency in buildings. *Energy and Buildings* 40(9), pp.1747–1754.

Dorfner, J. and Hamacher, T. 2014. Large-Scale District Heating Network Optimization. *IEEE Transactions on Smart Grid*, pp. 1–8.

Ekici, B.B. and Aksoy, U.T. 2009. Prediction of building energy consumption by using artificial neural networks. *Advances in Engineering Software* 40(5), pp. 356–362.

European Commission 2015. *Climate Action - Paris Agreement* [Online]. Available at: http://ec.europa.eu/clima/policies/international/negotiations/future/index_en.htm [Accessed: 12 January 2016].

Fazlollahi, S., Becker, G. and Maréchal, F. 2014. Multi-objectives, multi-period optimization of district energy systems: III. Distribution networks. *Computers & Chemical Engineering*.

Fazlollahi, S. and Marechal, F. (2013). Multi-objective, multi-period optimization of biomass conversion technologies using evolutionary algorithms and mixed integer linear programming (MILP). *Applied Thermal Engineering* 50(2), pp. 1504–1513.

Fernández-López, M. 1999. Overview of methodologies for building ontologies. In: *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods*. Sweden.

Fernandez, S., Rodriguez, F.J., Sanz, I., Mataix, C., Giron, C. and Moranchel, M. 2013. Multi-layer agent-based architecture for smart grid monitoring. In: *Proceedings of 2013 International Conference on Renewable Energy Research and Applications, ICRERA 2013*. IEEE Computer Society, pp. 143–148.

Fox, M.S., Barbuceanu, M., Gruninger, M. and Lin, J. 1997. An Organization Ontology for Enterprise Modelling. In: *Proceedings of ICEIMT '97, International Conference on Enterprise Integration and Modeling Technology.* Torino; Italy, 28–30 October, 1997.

Gadd, H. and Werner, S. 2014. Achieving low return temperatures from district heating substations. *Applied Energy* 136, pp. 59–67.

Georgievski, I., Nguyen, T.A. and Aiello, M. 2013. Combining activity recognition and AI planning for energy-saving offices. In: *Proceedings - IEEE 10th International Conference on Ubiquitous Intelligence and Computing, UIC 2013 and IEEE 10th International Conference on Autonomic and Trusted Computing, ATC 2013*. pp. 238–245.

Grassi, M., Nucci, M. and Piazza, F. 2011. Towards an ontology framework for intelligent smart home management and energy saving. In: *Proceedings - ISIE 2011: 2011 IEEE International Symposium on Industrial Electronics*. pp. 1753–1758.

Grüninger, M. and Fox, M.S. 1995. Methodology for the Design and Evaluation of Ontologies. In: *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95*.

Gudnason, G. and Scherer, R.J. 2012. *EWork and eBusiness in Architecture, Engineering and Construction Proceedings of the European Conference on Product and Process Modelling 2012, Reykjavik, Iceland, 25-27 July 2012*. Boca Raton: CRC Press.

Hamdy, M., Hasan, A. and Siren, K. 2011. Impact of adaptive thermal comfort criteria on building energy use and cooling equipment size using a multi-objective optimization scheme. *Energy and Buildings* 43(9), pp. 2055–2067.

Ham, Y. and Golparvar-Fard, M. 2013. EPAR: Energy Performance Augmented Reality models for identification of building energy performance deviations between actual measurements and simulation results. *Energy and Buildings* 63, pp.15–28.

Han, J., Jeong, Y.-K. and Lee, I. 2011. Efficient Building Energy Management System Based on Ontology, Inference Rules, and Simulation. In: *2011 International Conference on Intelligent Building and Management*.

Hashemi, R. 2009. A Developed Offline Model for Optimal Operation of Combined Heating and Cooling and Power Systems. *IEEE Transactions on Energy Conversion* 24(1), pp. 222–229.

Hippolyte, J.-L., Rezgui, Y., Haijiang, L. and Jayan, B. 2014. An ee-district ontology to support the development of the ee-District Information Model of the RESILIENT project. In: *EEBuilding Data Models : Energy Efficiency Vocabularies & Ontologies*. Nice: EEB Data Models Community, pp. 106–119.

Hiremath, R.B., Kumar, B., Balachandra, P. and Ravindranath, N.H. 2011. Decentralized sustainable energy planning of Tumkur district, India. *Environmental Progress & Sustainable Energy* 30(2), pp. 248–258.

Hlomani, H. and Stacey, D. 2014. Approaches, methods, metrics, measures, and subjectivity in ontology evaluation: A survey. *Semantic Web Journal*, 657. Available at: http://www.semantic-web-journal.net/system/files/swj657.pdf [Accessed: 12 January 2016].

Holland, J.H. 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. 1st MIT Press ed. Cambridge, Mass: MIT Press.

Hong, T., D'Oca, S., Turner, W.J.N. and Taylor-Lange, S.C. 2015. An ontology to represent energy-related occupant behavior in buildings. Part I: Introduction to the DNAs framework. *Building and Environment* 92, pp. 764–777.

Hou, S., Li, H. and Rezgui, Y. 2015. Ontology-based approach for structural design considering low embodied energy and carbon. *Energy and Buildings* 102, pp. 75–90.

Hou, S., Li, H. and Rezgui, Y. 2014. Ontology-based Semantic modelling of reinforced concrete structural design for building sustainability. In: *EG-ICE 2014, European Group for Intelligent Computing in Engineering - 21st International Workshop: Intelligent Computing in Engineering 2014*. Cardiff University.

Howell, S., Hippolyte, J.-L., Jayan, B., Reynolds, J. and Rezgui, Y. 2016. Web-based 3D Urban Decision Support through Intelligent and Interoperable Services. *Proceedings of the 2nd IEEE International Smart Cities Conference.* Trento; Italy, 12 September, 2016

Howell, S.K., Rezgui, Y. and Yuce, B. 2014. Knowledge-based holistic energy management of public buildings. In: Issa R.R. and Flood I. (eds.) *Computing in Civil and Building Engineering - Proceedings of the 2014 International Conference on Computing in Civil and Building Engineering*. American Society of Civil Engineers (ASCE), pp. 1667–1674.

Hsu, C.-C. and Wang, L.-Z. 2008. A smart home resource management system for multiple inhabitants by agent conceding negotiation. In: *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*. pp. 607–612.

Huber, M., Brust, S., Schütz, T., Constantin, A., Streblow, R. and Müller, D. 2015. Purely Agent Based Control of Building Energy Supply Systems. *Proceedings of ECOS 2015 - 28th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems.* Pau; France, June 30 - July 3, 2015.

Hu, J. and Olbina, S. 2011. Illuminance-based slat angle selection model for automated control of split blinds. *Building and Environment* 46(3), pp. 786–796.

Ihm, P. and Krarti, M. 2012. Design optimization of energy efficient residential buildings in Tunisia. *Building and Environment* 58, pp. 81–90.

Ikonen, E., Selek, I., Kovacs, J., Neuvonen, M., Szabo, Z., Bene, J. and Peurasaari, J. 2014. Short term optimization of district heating network supply temperatures. In: *Energy Conference (ENERGYCON), 2014 IEEE International*. IEEE, pp. 996–1003.

Inertia Consortium 2012. *INERTIA* [Online]. Available at: http://www.inertia-project.eu/inertia/index.html [Accessed: 25 March 2016].

International Electrotechnical Commission 2013. *Energy Management System Application Program Interface (EMS-API). Interface de Programmation D'application Pour Système de Gestion D'énergie (EMS-API). Part 301, Partie 301, Part 301, Partie 301,*. Geneva: International Electrotechnical Commission.

International Energy Agency 2009. *Cogeneration and District Energy*. Paris: International Energy Agency.

International Energy Agency (ed.) 2013. *Transition to Sustainable Buildings: Strategies and Opportunities to 2050*. Paris: IEA Publ.

Irani, Z., Jimenez, J., Lee, H., Martin, A., Molendini, V., Rascazzo, R., Savino, N., *et al.* 2015. A Decision Support System for Fostering Smart Energy Efficient Districts.

Jamot, J. and Olsson, F. 2013. Optimisation of the supply temperature on the southern district heating network in Lund. Available at: http://www.chemeng.lth.se/exjobb/E685.pdf [Accessed: 12 January 2016].

Jason Brownlee 2015. *Non-Dominated Sorting Genetic Algorithm - Clever Algorithms: Nature-Inspired Programming Recipes* [Online]. Available at:

http://www.cleveralgorithms.com/nature-inspired/evolution/nsga.html [Accessed: 21 August 2016].

Jayan, B., Li, H., Rezgui, Y., Hippolyte, J.-L. and Howell, S. 2016. An Analytical Optimization Model for Holistic Multiobjective District Energy Management - A Case Study Approach. *International Journal of Modeling and Optimization* 6(3), pp. 156–165.

Jayan, B., Li, H., Rezgui, Y., Hippolyte, J.L., Yuce, B., Yang, C. and Petri, I. 2014. An ontological approach to intelligent energy management in building. In: *EG-ICE 2014, European Group for Intelligent Computing in Engineering - 21st International Workshop: Intelligent Computing in Engineering 2014*. Cardiff University.

Jiang, X., Jing, Z., Li, Y., Wu, Q. and Tang, W. 2014. Modelling and operation optimization of an integrated energy based direct district water-heating system. *Energy* 64, pp. 375–388.

Jones, D., Bench-Capon, T. and Visser, P. 1998. Methodologies for ontology development. Available at: https://cgi.csc.liv.ac.uk/~tbc/publications/itknows.pdf [Accessed: 16 August 2016].

Kadolsky, M., Windisch, R. and Scherer, R.J. 2015. Knowledge management framework for monitoring systems improving building energy efficiency. In: *Environmental, Energy and Structural Monitoring Systems (EESMS), 2015 IEEE Workshop on*. IEEE, pp. 33–38.

Kalogirou, S.A. and Bojic, M. 2000. Artificial neural networks for the prediction of the energy consumption of a passive solar building. *Energy* 25(5), pp. 479–491.

Katranuschkov, P., Scherer, R.J., Baumgärtel, K., Hoch, R., Laine, T., Protopsaltis, B., Gudnason, G., *et al.* 2015. *Intelligent Services for Energy-Efficicent Design and Life Cycle Simulation.* Technische Universität Dresden, Institute of Construction Informatics.

Keirstead, J., Jennings, M. and Sivakumar, A. 2012. A review of urban energy system models: Approaches, challenges and opportunities. *Renewable and Sustainable Energy Reviews* 16(6), pp. 3847–3866.

Kim, T., Jeong, Y.-K. and Lee, I.-W. (2014). An approach for knowledge cloud service using ontology-based BEMS. In: *International Conference on ICT Convergence*. IEEE Computer Society, pp. 806–807. Available at: http://www.scopus.com/inward/record.url?eid=2-s2.0-84920191403&partnerID=40&md5=7a30aa09bdcbf8e6d5ae5b7b02c980c5 [Accessed: 22 October 2014].

Kofler, M.J., Iglesias-Vázquez, F. and Kastner, W. 2013. An ontology for representation of user habits and building context in future smart homes. In: de Wilde P., Suter G. and Rafiq Y. (eds.) *European Group for Intelligent Computing in Engineering, EG-ICE 2013 - 20th International Workshop: Intelligent Computing in Engineering*. Universiteit Twente.

Kofler, M.J. and Kastner, W. 2010. A knowledge base for energy-efficient smart homes. In: *2010 IEEE International Energy Conference and Exhibition, EnergyCon 2010*. pp. 85–90.

Kofler, M.J., Reinisch, C. and Kastner, W. 2011. An intelligent knowledge representation of smart home energy parameters. In: *Proceedings of the World Renewable Energy Congress (WREC 2011), Linköping, Sweden*.

Kofler, M.J., Reinisch, C. and Kastner, W. 2012. A semantic representation of energy-related information in future smart homes. *Energy and Buildings* 47, pp. 169–179.

Konak, A., Coit, D.W. and Smith, A.E. 2006. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety* 91(9), pp. 992–1007.

König, M. and Stankovski, V. 2012. An ontology of energy efficiency and renewable energy sources in buildings. In: *Proceedings, Annual Conference - Canadian Society for Civil Engineering*. pp. 183–192.

König, M., Truong, H.-L., Dustdar, S. and Stankovski, V. 2011. Information modelling for sustainable buildings. In: *ACM International Conference Proceeding Series*. pp. 507–510.

Kuriyan, K. 2015. *An Interactive Framework for Building and Analysing Models of Urban Energy Systems*. Elsevier.

Li, H.A. and Nair, N.-K.C. 2015. Multi-agent systems and demand response: A systematic review. In: *2015 Australasian Universities Power Engineering Conference (AUPEC)*, Wollongong, New South Wales, 27-30 September, 2015, IEEE. doi: 10.1109/AUPEC.2015.7324827.

Linked Data Tools 2015. *Tutorial 3: Semantic Modeling* [Online]. Available at: http://www.linkeddatatools.com/semantic-modeling [Accessed: 1 July 2016].

Liu, M., Shi, Y. and Fang, F. 2014. Combined cooling, heating and power systems: A survey. *Renewable and Sustainable Energy Reviews* 35, pp. 1–22.

López, G., Custodio, V., Moreno, J.I., Sikora, M., Moura, P. and Fernández, N. 2015. Modeling Smart Grid neighborhoods with the ENERsip ontology. *Computers in Industry* 70, pp. 168–182.

Lund, H., Werner, S., Wiltshire, R., Svendsen, S., Thorsen, J.E., Hvelplund, F. and Mathiesen, B.V. 2014. 4th Generation District Heating (4GDH). *Energy* 68, pp. 1–11.

Macek, K., Mařík, K. and Stluka, P. 2011. Ontology-Driven Design of an Energy Management System. In: *Computer Aided Chemical Engineering*. Elsevier, pp. 2009–2013.

Madrazo, L. and Sicilia, Á. 2014. *Deliverable 4.6 VoCamp 'Energy Data Models for Urban Planning'*. Barcelona: FUNITEC.

Madrazo, L., Sicilia, Á. and Gamboa, G. 2012. SEMANCO: Semantic tools for carbon reduction in urban planning. In: *eWork and eBusiness in Architecture, Engineering and*

*Construction - Proceedings of the European Conference on Product and Process Modelling 2012, ECPPM 2012*. pp. 899–908.

Maifredi, C., Puzzi, L. and Beretta, G.P. 2000. Optimal power production scheduling in a complex cogeneration system with heat storage. In: *Energy Conversion Engineering Conference and Exhibit, 2000. (IECEC) 35th Intersociety*. pp. 1004–1012 vol.2.

Mancarella, P. and Chicco, G. 2008. Assessment of the greenhouse gas emissions from cogeneration and trigeneration systems. Part II: Analysis techniques and application cases. *Energy* 33(3), pp. 418–430.

Mařík, K., Schindler, Z. and Stluka, P. 2008. Decision support tools for advanced energy management. *Energy* 33(6), pp. 858–873.

Marique, A.-F., Penders, M. and Reiter, S. 2013. From Zero Energy Building to Zero Energy Neighbourhood. In: *Proceedings of the 29th PLEA Conference. Munich*.

McMorran, A.W. 2007. *An Introduction to IEC 61970-301 and 61968-11: The Common Information Model*. Available at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.112.4648&rep=rep1&type=pdf [Accessed: 12 January 2016]

Morbach, J., Wiesner, A. and Marquardt, W. 2007. *A Meta Model for the Design of Domain Ontologies*. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.573.1774&rep=rep1&type=pdf [Accessed: 27 March 2016].

Mousavi, A. and Vyatkin, V. 2015. Energy efficient agent function block: A semantic agent approach to IEC 61499 function blocks in energy efficient building automation systems. *Automation in Construction* 54, pp. 127–142.

Murai, M., Sakamoto, Y. and Shinozaki, T. 1999. An optimizing control for district heating and cooling plant. In: *Control Applications, 1999. Proceedings of the 1999 IEEE International Conference on*. pp. 600–604 vol. 1.

Muthumanickam, A., Jain, R.K., Taylor, J.E. and Bulbul, T. 2014. Development of a novel BIM-energy use ontology. In: *Construction Research Congress 2014: Construction in a Global Network - Proceedings of the 2014 Construction Research Congress*. American Society of Civil Engineers (ASCE), pp. 150–159.

National Building Specification 2016. *Buidling Information Modelling* [Online]. Available at: http://www.thenbs.com/bim/what-is-bim.asp [Accessed: 24 January 2016].

Nemirovski, G., Nolle, A., Sicilia, Á., Ballarini, I. and Corado, V. 2013. Data integration driven ontology design, case study smart city. In: *ACM International Conference Proceeding Series*.

Neto, A.H. and Fiorelli, F.A.S. 2008. Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. *Energy and Buildings* 40(12), pp. 2169–2176.

Nguyen, T.A., Raspitzu, A. and Aiello, M. 2014. Ontology-based office activity recognition with applications for energy savings. *Journal of Ambient Intelligence and Humanized Computing* 5(5), pp. 667–681.

Niknam, M. and Karshenas, S. 2015. Sustainable Design of Buildings using Semantic BIM and Semantic Web Services. *Procedia Engineering* 118, pp. 909–917.

Noy, N.F. and McGuinness, D.L. 2001. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880.

Nucci, M., Grassi, M. and Piazza, F. 2013. *Ontology-Based Device Configuration and Management for Smart Homes*. Available at: http://www.scopus.com/inward/record.url?eid=2-s2.0-84879317233&partnerID=40&md5=ef3016d80695d3da21c074388ba9145e.

O'Donnell, J., See, R., Rose, C., Maile, T., Bazjanac, V. and Haves, P. 2011. SimModel: A domain data model for whole building energy simulation. In: *Proceedings of Building Simulation 2011: 12th Conference of International Building Performance Simulation Association*. pp. 382–389.

Openeering Multi-Objective Optimisation and Genetic Algorithms. [Online]. Available at: http://www.openeering.com/sites/default/files/Multiobjective_Optimization_NSGAII_0.pdf.

Pantaleo, A.M., Giarola, S., Bauen, A. and Shah, N. 2014a. Integration of biomass into urban energy systems for heat and power. Part I: An MILP based spatial optimization methodology. *Energy Conversion and Management* 83, pp. 347–361.

Pantaleo, A.M., Giarola, S., Bauen, A. and Shah, N. 2014b. Integration of biomass into urban energy systems for heat and power. Part II: Sensitivity assessment of main techno-economic factors. *Energy Conversion and Management* 83, pp 362–376.

Pauwels, P. and Terkaj, W. 2016. EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. *Automation in Construction* 63, pp. 100–133.

Pauwels, P., Törmä, S., Beetz, J., Weise, M. and Liebich, T. 2015. Linked Data in Architecture and Construction. *Automation in Construction* 57, pp. 175–177.

Pease, A. 2016. *The Suggested Upper Merged Ontology (SUMO) - Ontology Portal* [Online]. Available at: http://www.adampease.org/OP/ [Accessed: 1 July 2016].

Petri, I., Li, H., Rezgui, Y., Chunfeng, Y., Yuce, B. and Jayan, B. 2014a. A HPC based cloud model for real-time energy optimisation. *Enterprise Information Systems,* pp. 1–21.

Petri, I., Li, H., Rezgui, Y., Chunfeng, Y., Yuce, B. and Jayan, B. 2014b. A modular optimisation model for reducing energy consumption in large scale building facilities. *Renewable and Sustainable Energy Reviews,* pp. 990–1002.

Pini Prato, A., Strobino, F., Broccardo, M. and Parodi Giusino, L. 2012. Integrated management of cogeneration plants and district heating networks. *Applied Energy* 97, pp. 590–600.

Ploennigs, J., Dibowski, H., Roder, A., Kabitzsch, K. and Hensel, B. 2011. *D4.1 Ontology Specification for Model-Based ICT System Integration*. HESMOS consortium.

Reinisch, C., Kofler, M.J. and Kastner, W. 2010. ThinkHome: A smart home as digital ecosystem. In: *4th IEEE International Conference on Digital Ecosystems and Technologies - Conference Proceedings of IEEE-DEST 2010, DEST 2010*. pp. 256–261.

Resilient Project Consortium 2012. *Resilient Project* [Online]. Available at: http://www.resilient-project.eu/ [Accessed: 23 May 2014].

Rezgui, Y. (2007). Text-based domain ontology building using Tf-Idf and metric clusters techniques. *The Knowledge Engineering Review*, 22(4).

Rossello-Busquet, A., Brewka, L.J., Soler, J. and Dittmann, L. 2011. OWL Ontologies and SWRL Rules Applied to Energy Management. In: IEEE, pp. 446–450. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5754262 [Accessed: 21 May 2014].

Rossello-Busquet, A., Soler, J. and Dittmann, L. 2011. A novel home energy management system architecture. In: *Proceedings - 2011 UKSim 13th International Conference on Modelling and Simulation, UKSim 2011*. pp. 387–392.

Roussey, C., Pinet, F., Kang, M.A. and Corcho, O. 2011. An Introduction to Ontologies and Ontology Engineering. In: *Ontologies in Urban Development Projects*. London: Springer London, pp. 9–38.

Sakawa, M., Kato, K., Ushiro, S. and Inaoka, M. 2001. Operation planning of district heating and cooling plants using genetic algorithms for mixed integer programming. *Applied Soft Computing* 1(2), pp. 139–150.

Sallinen, M., Pulkkinen, T., Son, Y.-S., Park, J.-H. and Lee, Y.-H. 2012. Service platform for context reasoning in home environment. In: *International Conference on ICT Convergence*. Jeju; Korea, 15-17 October, 2012. IEEE.

Schellong, W. 2012. Energy efficiency and sustainability as a part of the engineering education. *Revista Congreso Universidad* 1(1), pp. 1–11.

Scherer, R.J., Katranuschkov, P., Kadolsky, M. and Laine, T. 2012. Ontology-based integration of building and related information models for integrated lifecycle energy management. In: *eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the European Conference on Product and Process Modelling 2012, ECPPM 2012*. pp. 951–956.

Schevers, H.A.J. and Drogemuller, R.M. 2005. Semantic Web for an Integrated Urban Software System. In: *MODSIM*. pp. 2040–2046. Available at: http://mssanz.org.au/modsim05/papers/schevers.pdf [Accessed: 23 May 2014].

Schiendorfer, A., Wautelet, Y. and Reif, W. 2015. Modeling hierarchical resources within a unified ontology a position paper. In: van den Herik J., Filipe J., Loiseau S., Filipe J. and Duval B. (eds.) *ICAART 2015 - 7th International Conference on Agents and Artificial Intelligence, Proceedings*. SciTePress, pp. 614–619.

Semanco Consortium 2011. *SEMANCO Semantic Tools for Carbon Reduction in Urban Planning* [Online]. Available at: http://semanco-project.eu/ [Accessed: 26 March 2016].

Sesame Project Consortium 2011. *Sesame Project* [Online]. Available at: https://www.sesame-project.eu/ [Accessed: 10 January 2016].

Shah, N., Chao, K.-M., Zlamaniec, T. and Matei, A. 2011. Ontology for Home Energy Management Domain. In: Cherifi, H., Zain, J. M. and El-Qawasmeh, E. (eds.) *Digital Information and Communication Technology and Its Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 337–347.

Shah, N., Tsai, C.-F., Chao, K.-M. and Lo, C.-C. 2010. Intelligent household energy management recomender system. In: *INNOV 2010 - Proceedings of the Multi-Conference on Innovative Developments in ICT*. pp. 51–56.

Sicilia, Á., Madrazo, L. and Costa, G. 2015. Building a semantic-based decision support system to optimize the energy use in public buildings: the OPTIMUS project. In: *Sustainable Places 2015*. Italy: Sigma Orionis.

Sicilia, Á., Madrazo, L. and Pleguezuelos, J. 2015. Integrating multiple data sources, domains and tools in urban energy models using semantic technologies. In: *eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the 10th European Conference on Product and Process Modelling, ECPPM 2014*. CRC Press/Balkema, pp. 837–844.

Sicilia, A., Nemirovski, G., Massetti, M. and Madrazo, L. 2015. The RÉPENER linked dataset. *Semantic Web* 6(2):131–137.

SportE2 Project Consortium 2014. *Energy Efficiency for Sport Facilities - SportE2* [Online]. Available at: http://www.sporte2.eu/ [Accessed: 9 December 2015].

Staab, S. and Studer, R. (eds.) 2009. *Handbook on Ontologies*. 2nd ed. Berlin: Springer.

Structured Dynamics 2014. *Ontology Best Practices - OSF Wiki* [Online]. Available at: http://wiki.opensemanticframework.org/index.php/Ontology_Best_Practices [Accessed: 2 July 2016].

Suárez-Figueroa, M., de Cea, G., Buil, C., Dellschaft, K., Fernandez-Lopez, M., Garcia, A., Gomez-Perez, A., *et al.* 2008. *D5.4.1. NeOn Methodology for Building Contextualized Ontology Networks*. NeON consortium.

Syberfeldt, A. 2014. Multi-Objective Optimization of a Real-World Manufacturing Process Using Cuckoo Search. In: Yang, X.-S. (ed.) *Cuckoo Search and Firefly Algorithm*. Cham: Springer International Publishing, pp. 179–193.

Tanasiev, V., Badea, A., Dinca, C. and Necula, H. 2015. *Intelligent System Concept for High-Energy Performance and Adaptable User Comfort*. Las Vegas, NV: Springer Verlag.

Terkaj, W. and Šojić, A. 2015. Ontology-based representation of IFC EXPRESS rules: An enhancement of the ifcOWL ontology. *Automation in Construction* 57, pp. 188–201.

Thorade, M., Rädler, J., Remmen, P., Maile, T., Wimmer, R., Cao, J., Lauster, M., *et al.* 2015. An Open Toolchain for Generating Modelica Code from Building Information Models, pp. 383–391. Available at: http://www.ep.liu.se/ecp_article/index.en.aspx?issue=118;article=41 [Accessed: 24 November 2015].

Tomašević, N.M., Batić, M.Č., Blanes, L.M., Keane, M.M. and Vraneš, S. 2015. Ontology-based facility data model for energy management. *Advanced Engineering Informatics* [Online]. Available at: http://linkinghub.elsevier.com/retrieve/pii/S1474034615000981 [Accessed: 24 November 2015].

Tomic, S., Fensel, A. and Pellegrini, T. 2010. SESAME demonstrator: Ontologies, services and policies for energy efficiency. In: *Proceedings of the 6th International Conference on Semantic Systems*. Graz; Austria, 1-3 September, 2010.

UCA International Users Group 2014. *Common Information Model (CIM) Users Group* [Online]. Available at: http://cimug.ucaiug.org/default.aspx/ [Accessed: 1 October 2016].

European Union 2010. *Directive 2010/31/EU Of The European Parliament and of the Council of 19 May 2010 on the Energy Performance of Buildings*. European Union. Available at: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2010:153:0013:0035:EN:PDF [Accessed: 20 February 2016].

Università Politecnica delle Marche 2014. *D 6.5 Pilot Final Report*. Italy.

University of Wisconsin *A Basic Introduction To Neural Networks* [Online]. Available at: http://pages.cs.wisc.edu/~bolo/shipyard/neural/local.html [Accessed: 21 August 2016].

Valiente-Rocha, P.A. and Lozano-Tello, A. 2010. Ontology-Based Expert System for Home Automation Controlling. In: García-Pedrajas, N., Herrera, F., Fyfe, C., Benítez, J. M. and Ali, M. (eds.) *Trends in Applied Intelligent Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 661–670.

Wang, F. and Prominski, M. (eds.) 2016. *Urbanization and Locality*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Wang, W., Rivard, H. and Zmeureanu, R. 2005. An object-oriented framework for simulation-based green building design optimization with genetic algorithms. *Advanced Engineering Informatics* 19(1), pp. 5–23.

Wernstedt, F., Davidsson, P. and Johansson, C. 2007. Demand side management in district heating systems. In: ACM Press, p. 1. Available at: http://portal.acm.org/citation.cfm?doid=1329125.1329454 [Accessed: 27 May 2014].

Wetter, M. and Wright, J. 2004. A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment* 39(8), pp. 989–999.

Wicaksono, H., Dobreva, P., Häfner, P. and Rogalski, S. 2015. *Methodology to Develop Ontological Building Information Model for Energy Management System in Building Operational Phase*. Springer Verlag.

Wicaksono, H., Dobreva, P., Häfner, P. and Rogalski, S. 2013. Ontology development towards expressive and reasoning-enabled building information model for an intelligent energy management system. In: *IC3K 2013; KEOD 2013 - 5th International Conference on Knowledge Engineering and Ontology Development, Proceedings*. pp. 38–47.

Wicaksono, H., Rogalski, S. and Kusnady, E. (2010). Knowledge-based intelligent energy management using building automation system. In: *2010 Conference Proceedings IPEC*. Singapore, 2010, pp. 1140-1145.

Wolosiuk, D., Ghiassi, N., Pont, U., Shayeganfar, F., Mahdavi, A., Fenz, S., Heurix, J., *et al.* 2014. *SEMERGY: Performance-Guided Building Design and Refurbishment within a Semantically Augmented Optimization Environment*. Bratislava: Trans Tech Publications.

World Wide Web Consortium 2008. *SPARQL Query Language for RDF* [Online]. Available at: https://www.w3.org/TR/rdf-sparql-query/#docOutline [Accessed: 2 July 2016].

World Wide Web Consortium *SPIN - FAQ* [Online]. Available at: http://spinrdf.org/faq.html [Accessed: 2 July 2016].

World Wide Web Consortium 2011. *SPIN - Modeling Vocabulary* [Online]. Available at: https://www.w3.org/Submission/spin-modeling/ [Accessed: 2 July 2016].

World Wide Web Consortium 2004. *SWRL: A Semantic Web Rule Language Combining OWL and RuleML* [Online]. Available at: https://www.w3.org/Submission/SWRL/ [Accessed: 2 July 2016].

Wright, J.A., Loosemore, H.A. and Farmani, R. 2002. Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and Buildings* 34(9), pp. 959–972.

Yalcintas, M. and Akkurt, S. 2005. Artificial neural networks applications in building energy predictions and a case study for tropical climates. *International Journal of Energy Research* 29(10), pp. 891–901.

Yang, C., Chen, B., Zhang, S., Li, Y. and Zhang, H. 2011. Design and implementation of general integrated optimization design software SiPESC.OPT. *Computer Aided Engineering* 20(4), pp. 42–48.

Yang, C., Li, H., Rezgui, Y., Petri, I., Yuce, B., Chen, B. and Jayan, B. 2014. High throughput computing based distributed genetic algorithm for building energy consumption optimization. *Energy and Buildings* 76, pp. 92–101.

Yang, J., Rivard, H. and Zmeureanu, R. 2005. On-line building energy prediction using adaptive artificial neural networks. *Energy and Buildings* 37(12), pp. 1250–1259.

Yezioro, A., Dong, B. and Leite, F. 2008. An applied artificial intelligence approach towards assessing building performance simulation tools. *Energy and Buildings* 40(4), pp. 612–620.

Yuce, B., Li, H., Rezgui, Y., Petri, I., Jayan, B. and Yang, C. 2014. Utilizing artificial neural network to predict energy consumption and thermal comfort level: An indoor swimming pool case study. *Energy and Buildings* 80, pp. 45–56.

Yuce, B. and Rezgui, Y. 2015. An ANN-GA Semantic Rule-Based System to Reduce the Gap Between Predicted and Actual Energy Consumption in Buildings. *IEEE Transactions on Automation Science and Engineering*:1–13.

Yuce, B., Rezgui, Y. and Howell, S.K. 2015. ANN-genetic algorithm-based rule generation for holistic energy management in public buildings. In: *eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the 10th European Conference on Product and Process Modelling, ECPPM 2014*. CRC Press/Balkema, pp. 401–407.

Zhao, H. and Magoulès, F. 2012. A review on the prediction of building energy consumption. *Renewable and Sustainable Energy Reviews* 16(6), pp. 3586–3592.

Zhou, Q., Natarajan, S., Simmhan, Y. and Prasanna, V. 2012. Semantic Information Modeling for Emerging Applications in Smart Grid. In: *2012 Ninth International Conference on Information Technology - New Generations*, Las Vegas, Nevada, 2012, pp. 775-782.

Zhou, Q., Simmhan, Y. and Prasanna, V. 2012. *Incorporating Semantic Knowledge into Dynamic Data Processing for Smart Power Grids*. Boston, MA.

# Appendix A – **REMO** Ontology classes, properties and rules.

## Classes of REMO ontology

1. **Classes related to building and energy consuming zones**

**EnergyConsumerRoom class**

Description

This class contains all individuals that are single rooms in buildings and consume energy. Building energy optimisation use cases can be applied to individuals of this class.

Relations

- EnergyConsumerRoom is a subclass of EnergyConsumerZone.
- It is disjoint with EnergyConsumerBuilding class.
- It cannot contain any subzone, which is defined by the statement '*hasSubzone* **exactly** 0'. This is because a room is considered to be the smallest zone.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<name of index class>_<name of building>_<name of energy consumer room>*.

Example: *EnergyConsumerBuilding_LeisureCentre_FitnessRoom*

**EnergyProducerBuilding class**

Description

This class contains all individuals which are energy producing buildings. Most districts have a central source of energy usually referred to as the energy centre or energy hub. This would, for example, be an individual of this class. It can also include any building which produces energy and supplies excess energy back to the main grid or to neighbouring buildings.

Relations

- EnergyProducerBuilding is a subclass of Building.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<name of index class>_<name of energy producer building>*.

Example: *EnergyProducerBuilding_EnergyCentre*

**`EnergyConsumerBuilding` class**

Description

This class contains all individuals which are energy consuming buildings. An energy consumer building can be an energy producing building as long as it satisfies the description of `EnergyProducerBuilding` class above.

Relations

- `EnergyConsumerBuilding` is a subclass of `Building`.
- `EnergyConsumerBuilding` is a subclass of `EnergyConsumerZone`. This is because sometimes the whole building can be controlled environmentally as one zone (see definition of `EnergyConsumerZone` below)

Usage of Class

The following naming convention is applied to name individuals of this class:
<*name of index class*>_<*name of energy consumer building*>.
Example: *EnergyConsumerBuilding_LeisureCentre*

2. **Classes related to energy sources.**

**`EnergySource_Centralised` class**

Description

This class contains all individuals that are central sources of energy to the district. An example of this would be the main electricity grid, which supplies electricity to a large area including the district considered.

Relations

- `EnergySource_Centralised` is a subclass of `EnergySource`.
- Disjoint with `EnergySource_Decentralised`.

**`EnergySource_Decentralised` class**

Description

This class contains all individuals that are energy-producing systems within the district or, in other words, are distributed energy resources (DER).

Relations

- `EnergySource_Decentralised` is a subclass of `EnergySource`.

- Disjoint with `EnergySource_Centralised`.

**`EnergySource_Decentralised_Electricity` class**

<u>Description</u>

This class contains all DERs which are sources of electricity to the district.

<u>Relations</u>

- `EnergySource_Decentralised_Electricity` is a subclass of `EnergySource_Decentralised`.

<u>Usage of Class</u>

This class seldom has individuals' instantiated because the subclasses of this class usually comes into play. In any case, the following naming convention is applied to name individuals of this class:

*<name of index class>_<name of decentralised electricity source>*

**`EnergySource_Decentralised_Heat` class**

<u>Description</u>

This class contains all DERs which are sources of heat to the district. Sometimes DERs can produce both heat and electricity so this class is not disjoint with `EnergySource_Decentralised_Electricity`.

<u>Relations</u>

- `EnergySource_Decentralised_Heat` is a subclass of `EnergySource_Decentralised`.

<u>Usage of Class</u>

This class seldom has individuals' instantiated because the subclasses of this class usually comes into play. In any case, the following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of decentralised heat source>*

**`NationalGrid` class**

<u>Description</u>

This class contains individuals if the district has a main electricity grid, through which electricity is provided.

Relations

- `NationalGrid` is a subclass of `EnergySource_Centralised`.
- Can have only one instance since most districts rely only on one major grid.
- Disjoint with `CombinedHeatPower`, `BiomassBoiler`, and `GasBoiler` classes

Usage of Class

The following naming convention is applied to name individuals of this class:

*<name of index class>_<name of district>*

For example: *NationalGrid_District7*

**`CombinedHeatPower` class**

Description

This class contains an individual which is a combined heat and power unit (CHP)

Relations

- This class is a subclass of `EnergySource_Decentralised_Heat`.
- This class is also a subclass of `EnergySource_Decentralised_Electricity`.
- Disjoint with `BiomassBoiler` and `GasBoiler` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<name of index class>_<name of CHP source>*

For example: *CombinedHeatPower_CHP1*

**`GasBoiler` class**

Description

This class contains an individual which is a gas boiler source

Relations

- This class is a subclass of `EnergySource_Decentralised_Heat`.

- Disjoint with `BiomassBoiler` and `CombinedHeatPower` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

<*name of index class*>_<*name of gas boiler source*>

For example: *GasBoiler_GB1*

**`BiomassBoiler` class**

Description

This class contains an individual which is a biomass boiler.

Relations

- This class is a subclass of `EnergySource_Decentralised_Heat`.
- Disjoint with `CombinedHeatPower` and `GasBoiler` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

<*name of index class*>_<*name of biomass boiler source*>

For example: *BiomassBoiler_BB1*

3. **Classes related to environmental and fuel properties**

**`Fuel_Type` class**

Description

This class contains individuals that represent the different fuel types in the district.

Relations

- This class is a subclass of `EnvironmentalAndFuelProperties`.
- It is equivalent to `eedistrict:FuelType` class.
- Disjoint with sibling classes: `Emissions_Transport`, `CalorificValue`, `Emissions_SpecificEmission`, and `Distance_BiomassSupplier`.

**`Emissions_Transport` class**

Description

This class contains individuals which represent the emissions due to transport of fuel. For example biomass fuel in most cases are delivered to site, and this therefore leads to transport emissions by the vehicle.

Relations

- This class is a subclass of `EnvironmentalAndFuelProperties`.
- Disjoint with sibling classes: `Distance_BiomassSupplier`, `CalorificValue`, `Emissions_SpecificEmission`, and `FuelType`.

Usage of Class

Instances of this class is named using the following convention:

*<Name of index class>_<Name of fuel>*

**`Distance_BiomassSupplier` class**

Description

This class contains individuals which represent the distance between the biomass fuel supplier and the district itself. This parameter is needed for calculation of emissions due to fuel transport. Biomass fuel, if used in the district, is usually transported from an external supplier.

Relations

- This class is a subclass of `EnvironmentalAndFuelProperties`.
- Disjoint with sibling classes: `Emissions_Transport`, `CalorificValue`, `Emissions_SpecificEmission`, and `FuelType`.

Usage of Class

Instances of this class is named using the following convention:

*<Name of index class>_<Name of district>*

For example: *Distance_BiomassSupplier_District7*

**`CalorificValue` class**

Description

This class contains individuals which represent the calorific value of the various fuels used in the district.

Relations

- This class is a subclass of `EnvironmentalAndFuelProperties`.
- Disjoint with sibling classes: `Distance_BiomassSupplier`, `Emissions_SpecificEmission`, `Emissions_Transport`, and `FuelType`.

Usage of Class

This class in the **REMO** ontology has two obvious instances already defined in the ontology as shown in figure 1 below:

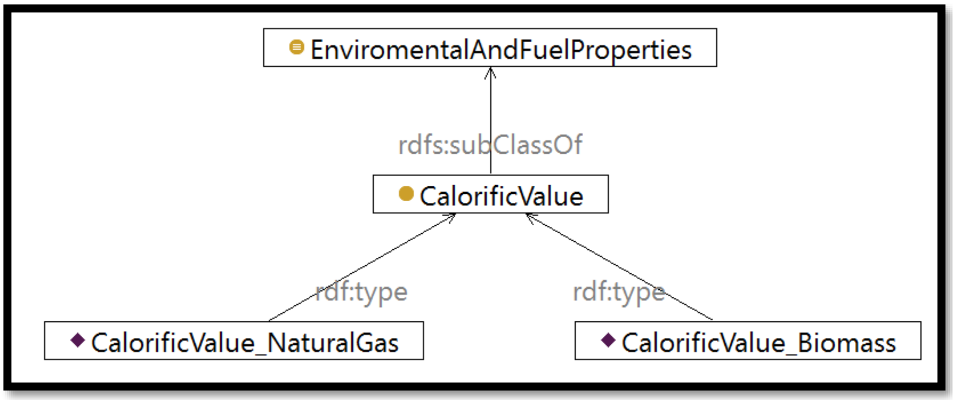



**Figure 1. `CalorificValue` class and its subclasses**

This is because most districts use biomass and natural gas as fuel. The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of fuel>*

**`Emissions_SpecificEmission` class**

Description

This class contains individuals which represent the specific emission of the various fuels used in the district.

Relations

- This class is a subclass of `EnvironmentalAndFuelProperties`.

- Disjoint with sibling classes: `Distance_BiomassSupplier`, `CalorificValue`, `Emissions_Transport`, and `FuelType`.

Usage of Class

As mentioned previously for class `CalorificValue`, the `Emissions_SpecificEmission` class also has two obvious instances defined in the **REMO** ontology as shown in figure 2 below:

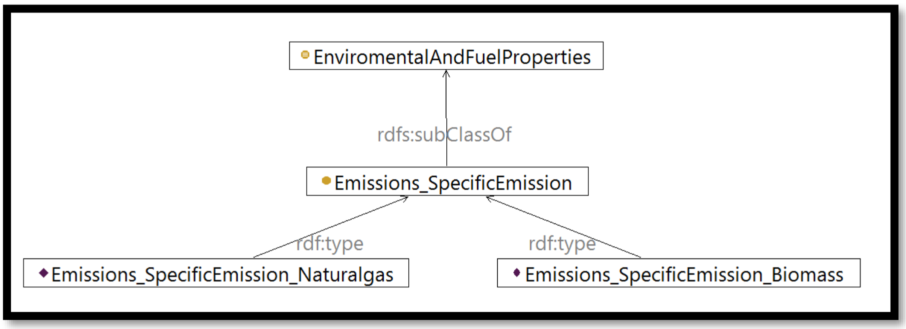



**Figure 2. `Emissions_SpecificEmission` class and instances.**

The following naming convention should be applied to any new individuals that are defined in this class:

*<Name of index class>_<Name of fuel>*

**`FuelType_Biomass` class**

Description

This class contains the biomass fuel type instances. There can be different types of biomass fuels as well in a district.

Relations

- This class is a subclass of `Fuel_Type`.
- Disjoint with `FuelType_Gas`.

Usage of Class

Instances of this class is named using the following convention:

*<Name of index class>_<Name of biomass fuel type>*

**`FuelType_Gas` class**

Description

This class contains the gas fuel type instances. For example, natural gas can be one of the instances under this class.

Relations

- This class is a subclass of `Fuel_Type`.
- Disjoint with `FuelType_Biomass`.

Usage of Class

Instances of this class is named using the following convention:

*<Name of index class>_<Name of gas fuel type>*

By default, both sub classes of FuelType are provided with instances in the **REMO** ontology as shown below in figure 3:

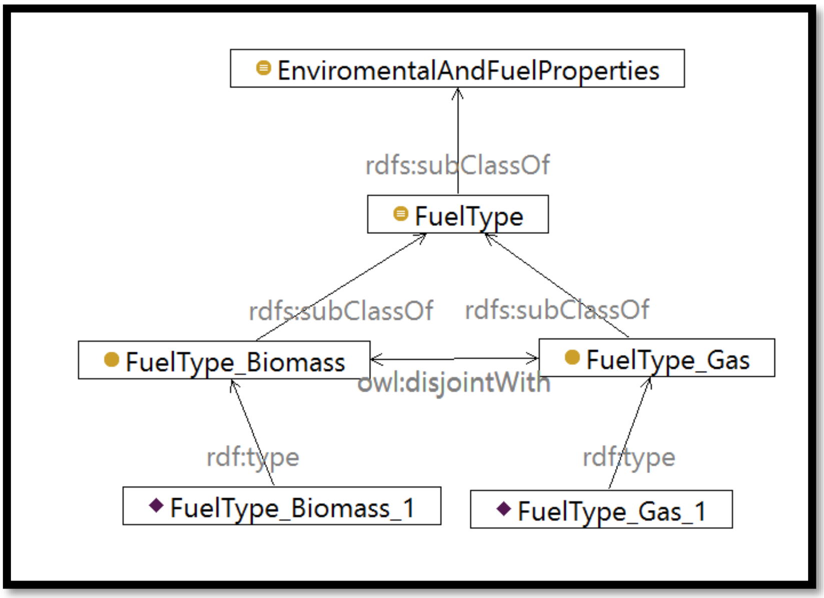



**Figure 3. `FuelType` classes and its subclasses.**

## 4. Classes related to energy source design parameters

### `Efficiency` class

Description

This class contains individuals that represent the efficiencies of the energy sources in the district.

Relations

- This class is a subclass of `EnergySourceDesignParameters`.
- It is equivalent to `eedistrict:Efficiency` class.

- Disjoint with sibling classes: `Elec2heatRatio` and `OutputPower`.

**`Elec2heatRatio` class**

Description

This class contains individuals that represent the heat to electricity ratio of energy sources, especially cogeneration systems (which produce both heat and electricity) such as CHP.

Relations

- This class is a subclass of `EnergySourceDesignParameters`.
- It is equivalent to `eedistrict:HeatPowerRatio` class.
- Disjoint with sibling classes: `Efficiency` and `OutputPower`.

Usage of Class

The following naming convention should be applied to any new individuals that need to be defined in this class:

*<Name of index class>_<Name of energy source linked to this property>*

For example: *Elec2heatRatio_CHP1*

**`Efficiency_HeatProduction` class**

Description

This class contains individuals which represent the efficiencies of the energy sources which produces heat energy in the district.

Relations

- This class is a subclass of `Efficiency.`
- Disjoint with sibling classes: `Efficiecny_HeatProduction.`

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of heat energy source whose efficiency parameter is defined here>*

For example: *Efficiency_HeatProduction_BiomassBoiler1*

**`Efficiency_ElectricityProduction` class**

Description

This class contains individuals which represent the efficiencies of the energy sources which produces electricity in the district.

Relations

- This class is a subclass of `Efficiency`.
- Disjoint with sibling classes: `Efficiecny_ElectricityProduction`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of electricity energy source whose efficiency parameter is defined here>*

For example: *Efficiency_ElectricityProduction_CHP*

**`OutputPower` class**

Description

This class contains individuals which represent the output power of the energy sources in the district.

Relations

- This class is a subclass of `EnergySourceDesignParameters`.
- Disjoint with sibling classes: `Elec2heatRatio`, and `Efficiency`.

**`OutputPower_Max` class**

Description

This class contains individuals which represent the maximum power of the energy sources.

Relations

- This class is a subclass of `OutputPower`.
- Disjoint with sibling classes: `OutputPower_Min`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of energy source whose maximum power is defined here>*

For example: *OutputPower_Max_CHP*

**`OutputPower_Min` class**

<u>Description</u>

This class contains individuals which represent the minimum power of the energy sources. For example, energy sources usually have a preferred lower output power, which can be about 30-50 % of the maximum power. This minimum power is defined in this class. Turning off energy sources over a short period of time might not always be feasible when it comes to real time energy management and hence this class is useful for production schedule optimisation.

<u>Relations</u>

- This class is a subclass of `OutputPower`.
- Disjoint with sibling classes: `OutputPower_Max`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of energy source whose minimum power is defined here>*

For example: *OutputPower_Min_CHP*

5. **<u>Classes related to building operational parameters</u>**

**`MeterReadings` class**

<u>Description</u>

This class contains individuals that represent the meter readings of the various buildings in the district. The Meter readings class is relevant for demand prediction and district energy optimisation as they represent the total energy demand of buildings, which will be optimised in real time using the demand optimisation use cases.

<u>Relations</u>

- This class is a subclass of `BuildingOperationalParameters`.
- It is equivalent to `socio_technical_systems:PhysicalProperty` class.
- Disjoint with sibling classes `Sensors` and `Actuators`.

### **`Sensors` class**

<u>Description</u>

This class contains individuals that represent the sensors of the various buildings in the district. This class is one of the most important classes in **REMO** ontology which is needed for demand side optimisation. The individuals in this class and its subclasses represent real-time dynamic values for the various parameters from the BMS systems of the various buildings.

<u>Relations</u>

- This class is a subclass of `BuildingOperationalParameters`.
- It is equivalent to `socio_technical_systems:PhysicalProperty` class
- Disjoint with sibling class `MeterReadings` and `Actuators`.

### **`Actuators` class**

<u>Description</u>

This class contains individuals that represent the actuators of the various energy systems found in buildings or districts. This class is crucial to the optimisation process as it represents the actuators in buildings which will be optimised in real time.

<u>Relations</u>

This class is a subclass of `BuildingOperationalParameters`.

1. Disjoint with sibling classes `Sensors` and `MeterReadings`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that need to be defined in this class:

*<Name of index class>*_*<Name of building>*_*<Name of actuator>*

**`MeterReadings_Heat` class**

Description

This class contains individuals which represent the heat meter readings of the buildings in the district.

Relations

- This class is a subclass of `MeterReadings`.
- Disjoint with sibling classes: `MeterReadings_Electricity`.

Usage of Class

This class in the **REMO** ontology has subclasses and the instances usually are defined in these subclasses (shown below). The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of heat meter>*

**`MeterReadings_Heat_Primary` class**

Description

This class contains individuals which represent the main heat meter readings of the buildings i.e. the total heat energy demand of the building is recorded through these meters.

Relations

- This class is a subclass of `MeterReadings_Heat`.
- Disjoint with sibling classes: `MeterReadings_Heat_Secondary`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of main heat meter>*

For example: M*eterReadings_Heat_Primary_LeisureCentre*

**`MeterReadings_Heat_Secondary` class**

Description

This class contains individuals which represent the sub-heat meter readings of the buildings i.e. the total heat energy demand for individual zones or sections or rooms in the building is recorded through these heat meters.

Relations

- This class is a subclass of `MeterReadings_Heat`.
- Disjoint with sibling classes: `MeterReadings_Heat_Primary`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of sub-heat meter>*

For example: M*eterReadings_Heat_Primary_LeisureCentreRoom1*

**`MeterReadings_Electricity` Class**

Description similar to `MeterReadings_Heat` class, but this class includes electricity meter readings rather than heat meter readings.

**`MeterReadings_Electricity_Primary` Class**

Description similar to `MeterReadings_Heat_Primary` class, but this class includes main electricity meter readings rather than main heat meter readings of buildings.

**`MeterReadings_Electricity_Secondary` Class**

Description similar to `MeterReadings_Heat_Secondary` class, but this class includes sub electricity meter readings rather than sub heat meter readings of buildings.

**`HumiditySensors` class**

Description

This class contains individuals which represent the humidity readings of a particular room or section or space in a building. It can also be outdoor humidity sensors.

Relations

- This class is a subclass of `Sensors`.
- Disjoint with sibling classes: `CarbonConcentrationSensors`, `OccupancySensors`, and `TempSensors`.

Usage of Class

This class in the **REMO** ontology has subclasses and the instances usually are defined in these subclasses (shown below). The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of humidity sensor>*

**`HumiditySensors_Indoor` class**

Description

This class contains individuals which represent the humidity readings of a particular room or space inside a building.

Relations

- This class is a subclass of `HumiditySensors`.
- Disjoint with sibling class `HumiditySensors_Outdoor`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of indoor humidity sensor>*

For example: *HumiditySensors_Indoor_LeisureCentre_FitnessRoom1*

**`HumiditySensors_Outdoor` class**

Description

This class contains individuals which represent the outdoor humidity readings of a building in the district.

Relations

- This class is a subclass of `HumiditySensors`.
- Disjoint with sibling class `HumiditySensors_Indoor`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of outdoor humidity sensor>*

For example: *HumiditySensors_Outdoor_LeisureCentre_SensorRHO*

**TempSensors class**

<u>Description</u>

This class contains individuals which represent the temperature readings of a particular room or space in a building. It can also be outdoor temperature sensors.

<u>Relations</u>

- This class is a subclass of `Sensors`.
- Disjoint with sibling classes: `CarbonConcentrationSensors`, `OccupancySensors`, and `HumiditySensors`.

<u>Usage of Class</u>

This class in the **REMO** ontology has subclasses and the instances usually are defined in these subclasses (shown below). The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of temperature sensor>*

**TempSensors_Indoor class**

<u>Description</u>

This class contains individuals which represent the temperature readings of a particular room or section or space inside a building.

<u>Relations</u>

- This class is a subclass of `TempSensors`.
- Disjoint with sibling class `TempSensors_Outdoor`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of indoor temperature sensor>*

For example: *TemSensors_Indoor_LeisureCentre_FitnessRoom1*

**TempSensors_Outdoor class**

<u>Description</u>

This class contains individuals which represent the outdoor temperature readings of a building in the district.

Relations

- This class is a subclass of `TempSensors`.
- Disjoint with sibling class `TempSensors_Indoor`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of outdoor temperature sensor>*

For example: *TempSensors_Outdoor_LeisureCentre_SensorTO*

**`OccupancySensors` class**

Description

This class contains individuals which represent the occupancy readings of a particular room or section or space in a building.

Relations

- This class is a subclass of `Sensors`.
- Disjoint with sibling classes: `CarbonConcentrationSensors`, `TempSensors`, and `HumiditySensors`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of occupancy sensor>*

For example: *OccupancySensors_LeisureCentre_Room1OccSensor*

**`CarbonConcetrationSensors` class**

Description

This class contains individuals which represent the carbon concentration readings of a particular room or section or space in a building. This sensor reading can be used for air handling unit optimisation. This parameter gives a good idea of the air quality in the space or room.

Relations

- This class is a subclass of `Sensors.`
- Disjoint with sibling classes: `OccupancySensors, TempSensors,` and `HumiditySensors.`

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of carbon concertation sensor>*

For example: *CarbonConcentrationSensors_LeisureCentre_Room1CCsensor*

6. **Classes related to BMS or EMS**

**`ParameterMapping_BMS` class**

Description

Individuals of this class represent endpoints of building related parameters. Using the associated properties of the individuals of this class, real-time dynamic values or historical data for these parameters can be retrieved.

Relations

- `ParameterMapping_BMS` is a subclass of `ParameterMapping`.
- Disjoint with `ParameterMapping_EMS` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>_<Name of parameter>*

For example: *ParameterMapping_BMS_LeisureCentre_CCRoom1*

**`ParameterMapping_EMS` class**

Description

Similar to ParameterMapping_BMS, endpoints of district-related parameters are represented by individuals of this class. Using the associated properties of the individuals of this class, real-time dynamic values or historical data for these parameters can be retrieved. They are mainly stored in the central energy management system (EMS) of the district.

Relations

- `ParameterMapping_EMS` is a subclass of `ParameterMapping`.
- Disjoint with `ParameterMapping_BMS` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of parameter>*

For example: *ParameterMapping_EMS_OutdoorTemperature*

7. **Classes related to district operational parameters.**

**`CrcTax` class**

Description

This class contains individuals that represent values needed for calculation of tax rate. The subclasses shown below in Figure 4 represent the most important tax parameters required for the calculations.

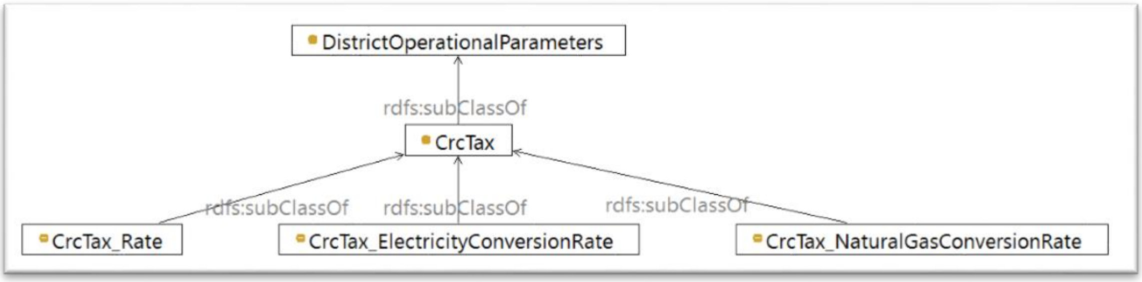



**Figure 4. `CrcTax` class and subclasses**

Relations

- This class is a subclass of `DistrictOperationalParameters`.
- Disjoint with sibling classes: `NetworkParameters`, `OperationalCosts`, and `OperationalSchedule`.

**`CrcTax_Rate` class**

Description

Contains value of the CRC tax rate which was a parameter used in the district analytical model as explained in Chapter 5 (Section 5.2.1).

Relations

- Subclass of `CrcTax` class.
- Disjoint with `CrcTax_ElectricityConversionRate` and `CrcTax_NaturalGasConversationRate` class.
- Equivalent to `eedistrict:TaxRate`

Usage of Class

The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of tax rate individual>*

For example: *CrcTax_Rate_District7*

**`CrcTax_ElectricityConversionRate` class**

Description

Individuals in this class is again used in the district analytical model.

Relations

- Subclass of `CrcTax` class.
- Disjoint with `CrcTax_Rate and` `CrcTax_NaturalGasConversationRate` class.
- Equivalent to `eedistrict:electricityConversionRate` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of electricity conversion rate parameter>*

For example: *CrcTax_ElectricityConversionRate_District7ElectricityConversionRate*

**`CrcTax_NaturalGasConversationRate` class**

Description

Individuals in this class is again used in the district analytical model.

Relations

- Subclass of `CrcTax` class.
- Disjoint with `CrcTax_Rate and` `CrcTax_ElectricityConversationRate` class.
- Equivalent to `eedistrict:naturalGasConversionRate` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of natural gas conversion rate parameter>*

For example: *CrcTax_NaturalGasConversationRate*

*_District7NaturalGasConversionRate*

### **`NetworkParameters` class**

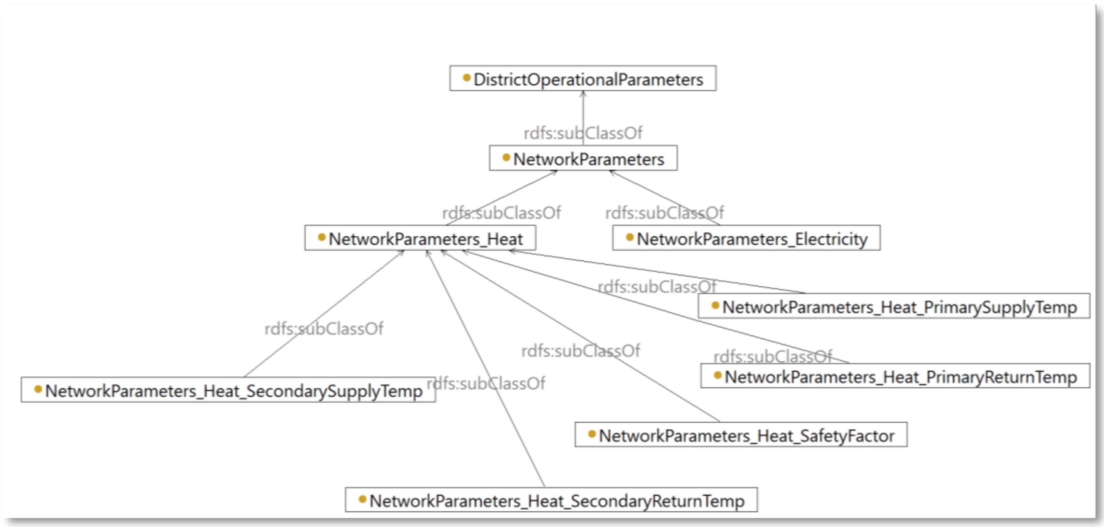



**Figure 5. `NetworkParameters` class and subclasses**

<u>Description</u>

This class contains individuals that represent values of the district network relevant to heat and electricity energy calculations; for example, supply temperature and return temperature at various points in the district network.

<u>Relations</u>

- This class is a subclass of `DistrictOperationalParameters`.
- Disjoint with sibling classes: `CrcTax`, `OperationalCosts`, and `OperationalSchedule`.

### **`NetworkParameter_Electricity` class**

<u>Description</u>

Contains individuals or subclasses that are related to electricity network parameters. For example, information on busbars of the district can be instantiated here if needed.

<u>Relations</u>

- Subclass of `NetworkParameter`.
- Disjoint with `NetworkParameter_Heat` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of parameter>*

**`NetworkParameter_Heat` class**

Description

Contains individuals or subclasses that are related to heat network parameters. For example information on a district heating network can be instantiated here. Some of this is also needed for district energy optimisation using the analytical model for example.

Relations

- Subclass of `NetworkParameter`.
- Disjoint with `NetworkParameter_Electricity` class.

Usage of Class

As shown in figure above, there are various subclasses and these classes usually has individuals instantiated. The `NetworkParameter_Heat` class acts mainly as a superclass with no or less individuals. The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heating parameter>*

**`NetworkParameters_Heat_PrimarySupplyTemp` class**

Description

Contains individuals related to primary supply temperature of district heating network. The parameter here refers to supply temperature to a particular building in the district or it can also be supply temperature provided by an energy producing source or building.

Relations

- Subclass of `NetworkParameter_Heat`.
- Disjoint with classes:
  `NetworkParameters_Heat_PrimaryReturnTemp`,
  `NetworkParameters_Heat_SecondarySupplyTemp`,

`NetworkParameters_Heat_SecondaryReturnTemp`, and
`NetworkParameters_Heat_SafetyFactor`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heat parameter>*

For example: *NetworkParameters_Heat_PrimarySupplyTemp_LeisureCentre.*

**`NetworkParameters_Heat_PrimaryReturnTemp` class**

Description

Contains individuals related to primary return temperature of a district heating network. The parameter here refers to return temperature from a particular building in the district heating network. Sometimes it can be the return temperature of water which is sent to a heat producing source or building.

Relations

- Subclass of `NetworkParameter_Heat`.
- Disjoint with classes:
  `NetworkParameters_Heat_PrimarySupplyTemp`,
  `NetworkParameters_Heat_SecondarySupplyTemp`,
  `NetworkParameters_Heat_SecondaryReturnTemp`, and
  `NetworkParameters_Heat_SafetyFactor`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heat parameter>*

For example: *NetworkParameters_Heat_PrimaryReturnTemp_LeisureCentre.*

**`NetworkParameters_Heat_SecondarySupplyTemp` class**

Description

Contains individuals related to Secondary supply temperature in a particular building. The parameter here refers to supply temperature within a particular building which is dedicated for a particular area of the building. For example, domestic hot water network can be linked to a secondary heating network within the building and it has another dedicated heat exchanger for its application. This sometimes is branched from the primary heating network in the building i.e. it is branched off the primary heat exchanger of the building.

Relations

- Subclass of `NetworkParameter_Heat`.
- Disjoint with classes: `NetworkParameters_Heat_PrimaryReturnTemp`, `NetworkParameters_Heat_PrimarySupplyTemp`, `NetworkParameters_Heat_SecondaryReturnTemp`, and `NetworkParameters_Heat_SafetyFactor`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heat parameter>*

For example: *NetworkParameters_Heat_SecondarySupplyTemp_LeisureCentre.*

**NetworkParameters_Heat_SecondaryReturnTemp class**

Description

This instantiates individuals which contain the return temperature of the secondary network within the building.

Relations

- Subclass of `NetworkParameter_Heat`.
- Disjoint with classes: `NetworkParameters_Heat_PrimarySupplyTemp`, `NetworkParameters_Heat_SecondarySupplyTemp`, `NetworkParameters_Heat_PrimaryReturnTemp`, and `NetworkParameters_Heat_SafetyFactor`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heat parameter>*

For example: *NetworkParameters_Heat_SecondaryReturnTemp_LeisureCentre.*

**NetworkParameters_Heat_SafetyFactor class**

Description

Heat loss within the district heating network is important to be calculated. In many cases, detailed real time monitoring or advanced simulation models might not be available and hence this heat loss factor needs to be approximated. The NetworkParameters_Heat_SafetyFactor class contains instances which stores this value. This can also be a dynamic value that the EMS of the district computes. It can be approximated using simple historical data analysis. This value needs is taken into account by the analytical model and district optimisation model for optimising the heat production schedules in the district.

Relations

- Subclass of NetworkParameter_Heat.
- Disjoint with classes: NetworkParameters_Heat_PrimarySupplyTemp, NetworkParameters_Heat_SecondarySupplyTemp, NetworkParameters_Heat_PrimaryReturnTemp, and NetworkParameters_Heat_SecondaryReturnTemp.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of network heat safety factor>*

For example: *NetworkParameters_Heat_SafetyFactor_District7*

**ActivitySchedule class**

Description

Contains individuals that represents a particular activity schedule which can be building or district related. For example, a fitness room in the building might have activities planned throughout the week that can help calculate the average MET produced by users. This is useful for PMV (Fanger's comfort factor) calculation, and consequently, comfort optimisation. The individuals of this class represents these schedules and its actual dynamic values can be retrieved from the BMS or EMS through the corresponding ParameterMapping class individuals.

Relations

- Subclass of DistrictOperationalSchedule.

- Disjoint with classes: `ProductionScheduleHeat`, `ForecastedWeatherSchedule` and `DemandSchedule`.

Usage of Class

The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of activity schedule>*

*For example: ActivitySchedule_LeisureCentreFitnessRoomMET*

**`DemandSchedule` class**

Description

Contains individuals or subclasses that defines the twenty-four-hour demand profiles of buildings or spaces/rooms in the building. The individuals in this class is needed both for building optimisation and district optimisation. The various subclasses shown in figure 6 below shows the variety of schedules which is needed for real time energy management. Each of them are explained below in this section.

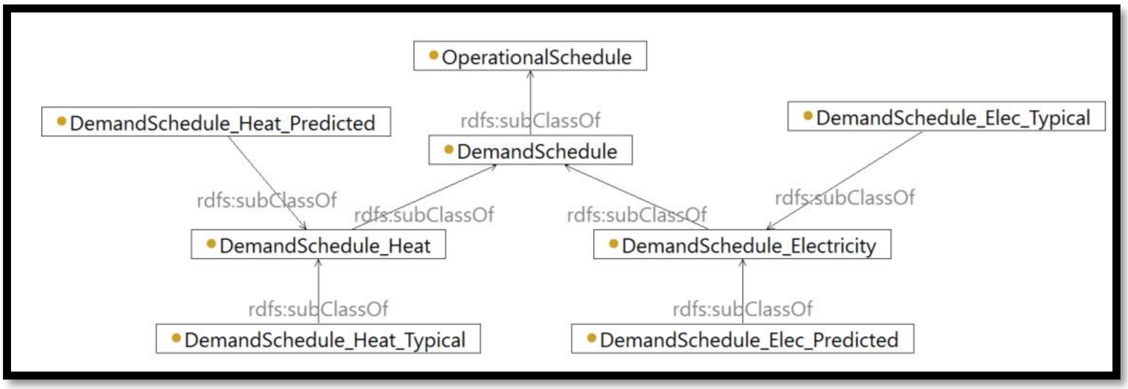



**Figure 6. `DemandSchedule` class and its subclasses**

Relations

- Subclass of `DistrictOperationalSchedule`.
- Disjoint with classes: `ProductionScheduleHeat`, `ForecastedWeatherSchedule` and `ActivitySchedule`.

Usage of Class

As shown as figure above, there are various subclasses to this class and these classes usually are instantiated. The `DemandSchedule` class acts mainly as a superclass with no or less individuals. The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of demand schedule>*

**`DemandSchedule_Heat` class**

Description

Contains individuals or subclasses that defines the twenty-four-hour heat demand profiles of buildings or spaces/rooms in the building.

Relations

- Subclass of `DemandSchedule` class.
- Disjoint with `DemandSchedule_Electricity` class.

Usage of Class

The subclasses to this class are usually are instantiated. The `DemandSchedule_Heat` class acts mainly as a superclass with no or less individuals. The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of heat demand schedule>*

**`DemandSchedule_Heat_Typical` class**

Description

A typical heat demand schedule is stored in individuals of this class. The individuals represent the schedule and the actual values for the schedule can be retrieved from the BMS itself through the corresponding individuals of `ParameterMapping_BMS` class. This typical schedule is derived from historical data and used for district optimisation when predicted heat demand schedules are not available.

Relations

- Subclass of `DemandSchedule_Heat` class.
- Disjoint with `DemandSchedule_Heat_Predicted` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of typical heat demand schedule>*

For example: *DemandSchedule_Heat_Typical_Building1*

**`DemandSchedule_Heat_Predicted` class**

Description

The individuals of this class is similar to individuals of `DemandSchedule_Heat_Typical` class. However, these are predicted demand schedules which are results of running the ANN models of **REMO** framework. These are used for district energy optimisation for day to day operations.

Relations

- Subclass of `DemandSchedule_Heat` class.
- Disjoint with `DemandSchedule_Heat_Typical` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of typical heat demand schedule>*

For example: *DemandSchedule_Heat_Predicted_Building1*

### DemandSchedule_Electricity Class

Description similar to `DemandSchedule_Heat` class, but this class includes electricity demand schedules rather than heat demand schedules.

### DemandSchedule_Electricity_Typical Class

Description similar to `DemandSchedule_Electricity_Typical` class, but this class includes typical electricity demand schedules rather than typical heat demand schedules.

### DemandSchedule_Elecctricity_Predicted Class

Description similar to `DemandSchedule_Electricity_Predicted` class, but this class includes predicted electricity demand schedules rather than predicted heat demand schedules.

### ProductionScheduleHeat class

Description

Contains individuals or subclasses that defines the twenty-four hour heat production profiles of buildings or energy sources in the building. This is needed mainly for district energy optimisation using the analytical model. Contains two subclasses as shown below in figure.

Note: Disjoint relationship between the two subclasses is not shown in figure 7 below.

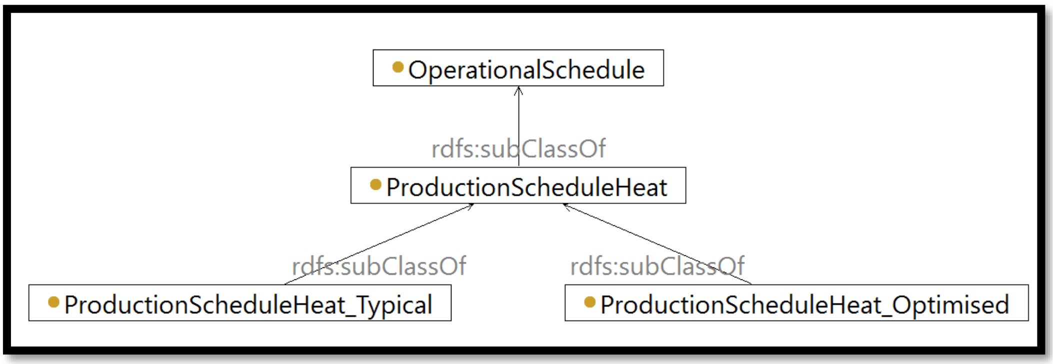



**Figure 7. `ProductionScheduleHeat` Class and its subclasses**

<u>Relations</u>

- Subclass of `DistrictOperationalSchedule` class.
- Disjoint with `DemandSchedule`, `ForecastedWeatherSchedule` and `ActivitySchedule` class.

<u>Usage of Class</u>

As shown in figure above, there are two subclasses to this class which are usually instantiated. The `ProductionScheduleHeat` class acts mainly as a superclass with no or less individuals. The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of production schedule>*

**`ProductionScheduleHeat_Typical` class**

<u>Description</u>

A typical heat production schedule is stored in these instances. This typical schedule is considered not to be optimised and is used when optimised production schedules are not available.

<u>Relations</u>

- Subclass of `ProductionScheduleHeat` class.
- Disjoint with `ProductionScheduleHeat_Optimised` class.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of typical production schedule>*

For example: *ProductionScheduleHeat_Typical_District7Schedule*

**`ProductionScheduleHeat_Optimised` class**

<u>Description</u>

The optimised heat production schedules of the district are represented by individuals of this class. The optimised production schedule is calculated using the analytical model and district optimisation model.

<u>Relations</u>

- Subclass of `ProductionScheduleHeat` class.
- Disjoint with `ProductionScheduleHeat_Typical` class.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of optimised heat production schedule>*

For example: *ProductionScheduleHeat_Optimised_District7Schedule*

**`ForecastedWeatherSchedule` class**

<u>Description</u>

Contains individuals or subclasses that defines the twenty-four-hour day ahead predictions for weather parameters. This is needed mainly for demand predictions of building using the ANN model.

Note: Disjoint relationship between the two subclasses is not shown in figure 8 below.

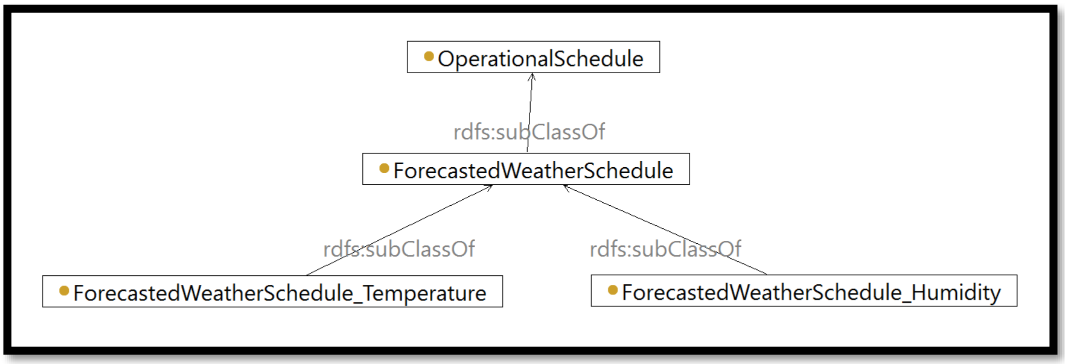



**Figure 8. `ForecastedWeatherSchedule` and its subclasses.**

<u>Relations</u>

- Subclass of `DistrictOperationalSchedule` class.

- Disjoint with `DemandSchedule`, `ProductionScheduleHeat` and `ActivitySchedule` class.

Usage of Class

As shown in figure above, it has two subclasses- `ForecastedWeatherSchedule_Temperature` represents the outdoor temperature prediction and `ForecastedWeatherSchedule_Humidity` represents the outdoor humidity prediction. These subclasses are usually instantiated and the following naming convention is applied to individuals:

*<Name of index class>_<Name of forecasted weather schedule>*

### `OperationalCosts` class

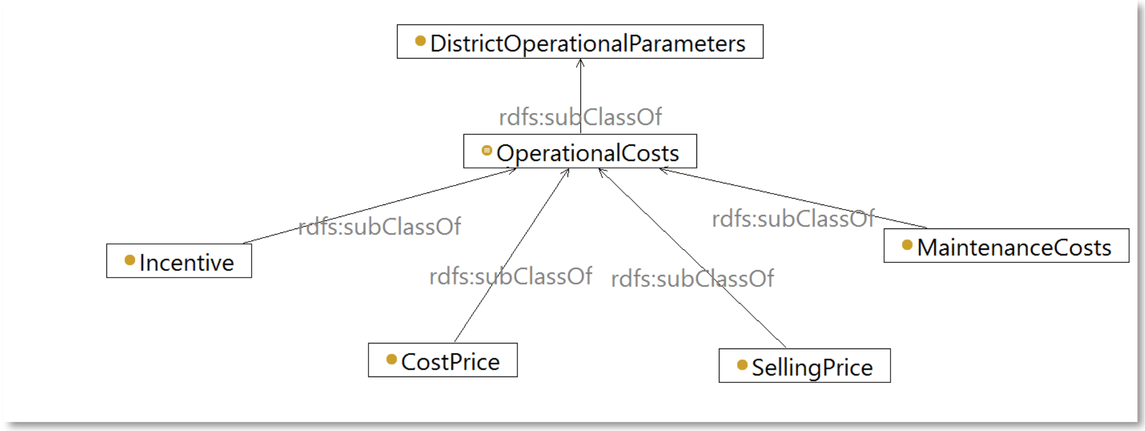



**Figure 9. `OperationalCosts` class and its subclasses**

Description

This class contains individuals that represent cost-related parameters of the district. Cost prices, selling prices, incentives, and maintenance costs are all subclasses of this class. They are relevant for the analytical model and optimisation model for supply side optimisation.

Relations

- This class is a subclass of `DistrictOperationalParameters`.
- Disjoint with sibling classes: `CrcTax`, `NetworkParameters`, and `OperationalSchedule`.

**`MaintenanceCosts` class**

Description

Contains individuals or subclasses that are related to maintenance costs of energy sources systems.

Relations

- Subclass of `OperationalCosts`.
- Disjoint with sibling classes: `CostPrice`, `Incentive`, and `SellingPrice`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of energy source>*

For example: *MaintenanceCosts_CHP*

**`Incentive` class**

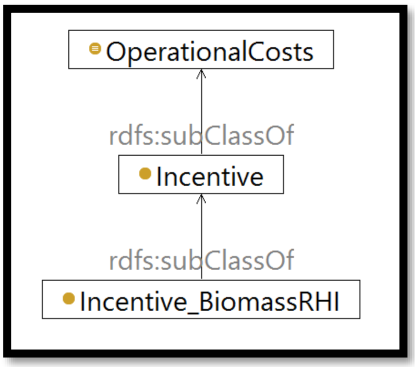



**Figure 10. `Incentive` class and its subclasses.**

Description

Contains individuals or subclasses that are related to incentives received in the district. Incentives are provided mainly for using low carbon sources or renewable energy sources. For example, using biomass boilers can gain incentives.

Relations

- Subclass of `OperationalCosts`.
- Disjoint with sibling classes: `CostPrice`, `MaintenanceCosts`, and `SellingPrice`.

Usage of Class

Individuals are defined in its subclasses. In any case, the following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of incentive parameter>*

**`Incentive_BiomassRHI` class**

<u>Description</u>

This class contains individuals which represents the incentive received for unit energy produced using biomass boilers

<u>Relations</u>

- Subclass of `Incentive` class.

<u>Usage of Class</u>

The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of energy source>*

For example: *Incentive_BiomassRHI_BiomassBoiler*

**`CostPrice` class**

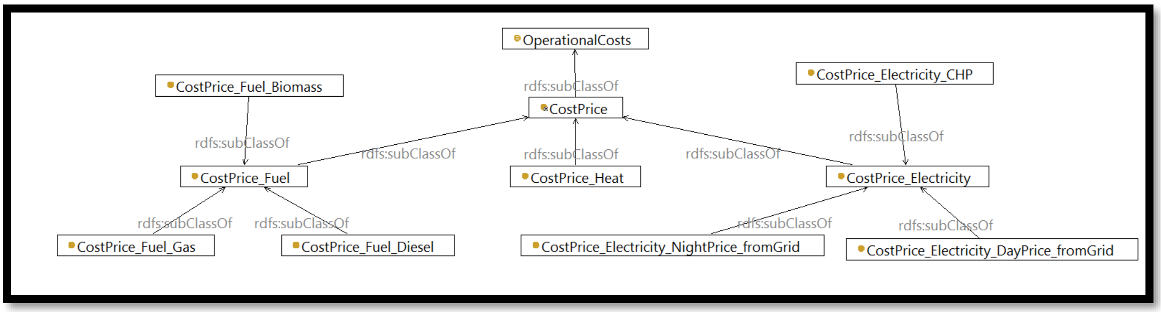



**Figure 11. `CostPrice` class and its subclasses.**

<u>Description</u>

Contains individuals or subclasses that are related to cost prices in the district. Subclasses include individuals representing cost prices for electricity, heat and the various fuel used in the district.

<u>Relations</u>

- Subclass of `OperationalCosts.`
- Disjoint with sibling classes: `Incentive`, `MaintenanceCosts`, and `SellingPrice`.

Usage of Class

As shown in figure above, various subclasses exist and individuals are usually defined here. The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of cost price parameter>*

**`CostPrice_Electricity` class**

Description

This class contains individuals which represents the cost prices related to electricity in the district

Relations

- Subclass of `CostPrice`.
- Disjoint with sibling classes: `CostPrice_Fuel` and `CostPrice_Heat`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of cost price parameter related to electricity purchase>*

**`CostPrice_Electricity_CHP` class**

Description

This class contains individuals which represents the purchase price of unit of electricity produced from a CHP source.

Relations

- Subclass of `CostPrice_Electricity`.
- Disjoint with sibling classes: `CostPrice_Electricity_DayPrice_Grid` and `CostPrice_Electricity_NightPrice_Grid`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of parameter>*

**`CostPrice_Electricity_DayPrice_Grid` Class**

Description

This class contains individuals which represents the purchase price of unit of electricity during the day, from a centralised source such as the main national grid. Main grid purchase price of electricity can vary throughout the day. The case study for which the ontology is built at the moment assumes that purchase price during hours of day is different to the price in the night.

Relations

- Subclass of `CostPrice_Electricity`.
- Disjoint with sibling classes: `CostPrice_Electricity_DayPrice_Grid` and `CostPrice_Electricity_CHP. I`

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of parameter or name of district>*

For example: *CostPrice_Electricity_DayPrice_Grid_District7*

**`CostPrice_Electricity_NightPrice_Grid` Class**

Description

This class contains individuals which represents the night-time purchase price per unit of electricity.

Relations

- Subclass of `CostPrice_Electricity`.
- Disjoint with sibling classes: `CostPrice_Electricity_DayPrice_Grid` and `CostPrice_Electricity_CHP`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of parameter or name of the district>*

For example: *CostPrice_Electricity_NightPrice_Grid_District7*

**`CostPrice_Fuel` class**

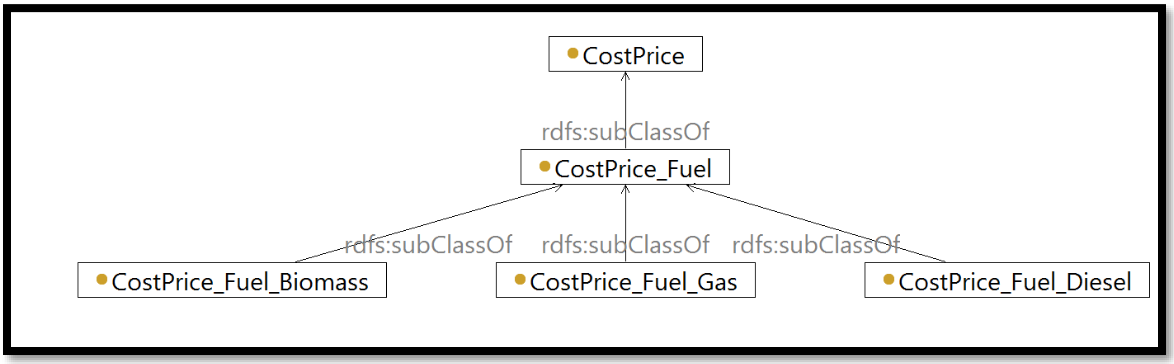



**Figure 12. `CostPrice_Fuel` class and its subclasses.**

Description

This class contains individuals which represents the cost prices of the different types of fuel used in the district. Figure 12 above shows the various subclasses involved. These subclasses are all mutually disjoint with each other.

Relations

- Subclass of `CostPrice`.
- Disjoint with sibling classes: `CostPrice_Electricity` and `CostPrice_Heat`.

Usage of Class

The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of cost price parameter related to fuel purchase>*

**`CostPrice_Fuel_Biomass` class**

Description

This class contains individuals which represents the purchase price of biomass fuel used in the district.

Relations

- Subclass of `CostPrice_Fuel`.
- Disjoint with sibling classes: `CostPrice_Fuel_Diesel` and `CostPrice_Fuel_Gas`

Usage of Class

The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of biomass fuel purchase price parameter>*

For example: *CostPrice_Fuel_Biomass_PurchasePriceDistrict7*

**`CostPrice_Fuel_Diesel` class**

Description similar to `CostPrice_Fuel_Biomass` class, but this class includes individuals that represent cost price of diesel fuel used in the district.

**`CostPrice_Fuel_Gas` class**

Description similar to `CostPrice_Fuel_Biomass` class, but this class includes individuals that represent cost price of natural gas used in the district.

**`CostPrice_Heat` class**

<u>Description</u>

This class contains individuals which represents the cost price of heat being purchased in the district. Usually this is from the district heating network.

<u>Relations</u>

- Subclass of `CostPrice` class
- Disjoint with sibling classes: `CostPrice_Electricity` and `CostPrice_Heat`.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of cost price parameter related to heat purchase>*

For example: *CostPrice_Heat_District7HeatNetwork*

**`SellingPrice` class**

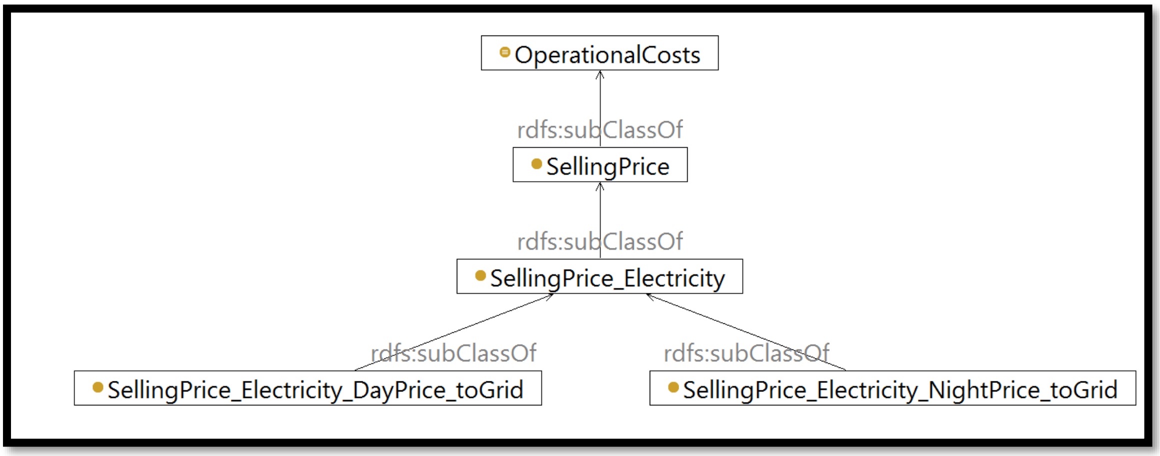



**Figure 13. `SellingPrice` class and its subclasses.**

Description

Contains individuals or subclasses that are related to selling prices in the district. Subclasses include individuals representing selling prices, especially for electricity, as excess production can be sent/sold back to grid.

Relations

- Subclass of `OperationalCosts`.
- Disjoint with sibling classes: `Incentive`, `MaintenanceCosts`, and `CostPrice`.

Usage of Class

As shown in figure above, two subclasses exist and individuals are usually defined in these. The following naming convention is applied to individuals of this class:

*<Name of index class>_<Name of selling price parameter>*

**`SellingPrice_Electricity` class**

Description

This class contains individuals which represents the selling prices related to electricity in the district.

Relations

- Subclass of `SellingPrice`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of selling price parameter for electricity sold to grid>*

**`SellingPrice_Electricity_DayPrice_toGrid` Class**

<u>Description</u>

This class contains individuals which represents the selling price of unit of electricity during the day, to the main national grid. Selling price of electricity can vary throughout the day. The ontology here assumes that selling price during hours of day is different to the price in the night, similar to the assumption made in the case of purchase prices, mentioned earlier under `CostPrice_Electricity` class.

<u>Relations</u>

- Subclass of `SellingPrice_Electricity`.
- Disjoint with `SellingPrice_Electricity_NightPrice_toGrid` class.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of selling price parameter or name of district>*

For example: *SellingPrice_Electricity_DayPrice_toGrid_District7*

**`SellingPrice_Electricity_NightPrice_toGrid` Class**

<u>Description</u>

This class contains individuals which represents the selling price of unit of electricity during the night, to the main national grid.

<u>Relations</u>

- Subclass of `SellingPrice_Electricity`.
- Disjoint with `SellingPrice_Electricity_DayPrice_toGrid` class.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of selling price parameter or name of district>*

For example: *SellingPrice_Electricity_NightPrice_toGrid_District7*

8. **Classes related to use cases and scenarios for real-time energy management.**

**`UseCases_Building_Optimisation_ScenarioSwimmingPoolAhuOptimisation` class.**

Description

This class contains the optimisation related use cases applied within a building, especially looking into optimisation of air handling unit of a zone (or room) or space containing a swimming pool.

Relations

- Subclass of `UseCases_Building_Optimisation`.
- Disjoint with `UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>_<name of swimming pool ahu optimisation use case>*

For example: *UseCases_Building_Optimisation_ScenarioSwimmingPoolAhuOptimisation_LeisureCentre_SwimmingPoolArea.*

**`UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation` class.**

Description

This class contains the optimisation related use cases applied within a building, especially looking into optimisation of air handling unit of a zone (or room) or space.

Relations

- Subclass of `UseCases_Building_Optimisation`.
- Disjoint with `UseCases_Building_Optimisation_ScenarioSwimmingPoolAhuOptimisation` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>_<name of room AHU optimisation use case>*

For example: *UseCases_Building_Optimisation_ScenarioRoomAhuOptimisation_LeisureCentre_FitnessRoom.*

**`UseCases_Building_Prediction_Model_OverallDemandProfile_Heat` Class**

<u>Description</u>

Individuals of this class represent use cases that supports running of ANN models which predicts the overall heat demand profile of buildings.

<u>Relations</u>

- Subclass of `UseCases_Building_Prediction_OverallDemandProfile`.
- Disjoint with sibling class `UseCases_Building_Prediction_Model_OverallDemandProfile_Electricity`.

<u>Usage of Class</u>

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>*

For example:

*UseCases_Building_Prediction_OverallDemandProfile_Heat_LeisureCentre*

**`UseCases_Building_Prediction_Model_OverallDemandProfile_Electricity` Class**

Similar to class `UseCases_Building_Prediction_Model_OverallDemandProfile_Heat`, but here the focus is on electricity demand profiles prediction of buildings and not heat.

**`UseCases_Building_Prediction_SwimmingPoolAhu` class.**

<u>Description</u>

This class contains the prediction related use cases applied within a building for developing and training an ANN model. This use case class is needed to support the `UseCases_Building_Optimisaiton_ScenarioSwimmingPoolAhuOptimisation` class, as it provides information for the training of the ANN model which is to be used for optimisation of air handling units in rooms or zones containing swimming pools. Therefore, the individuals of this class represent use cases which are applied to rooms or zones which has a swimming pool and requires AHU optimisation.

Relations

- Subclass of `UseCases_Building_Prediction_Training`.
- Disjoint with `UseCases_Building_Prediction_RoomAhu` class and `UseCases_Building_Prediction_OverallDemandProfile` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>_<name of swimming pool room prediction parameter use case>*

For example:
*UseCases_Building_Prediction_SwimmingPoolAhu__LeisureCentre_SwimmingPoolRoom1ANN.*

**UseCases_Building_Prediction_RoomAhu class.**

Description

Similar to its sibling class, this class has individuals which represent use cases class needed to support the `UseCases_Building_Optimisaiton_ScenarioRoomAhuOptimisation` class, as it provides information for the training of the ANN model which is to be used for optimisation of air handling units in rooms. Therefore, this class has individuals which represents use cases which are applied to rooms or zones which requires AHU optimisation.

Relations

- Subclass of `UseCases_Building_Prediction_Training`.

- Disjoint with `UseCases_Building_Prediction_SwimmingPoolAhu` class and `UseCases_Building_Prediction_OverallDemandProfile` class.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>_<name of room prediction parameter use case>*

For example: *UseCases_Building_Prediction_RoomAhu__LeisureCentre_FitnessRoomANN.*

**`UseCases_Building_Prediction_OverallDemandProfile` class.**

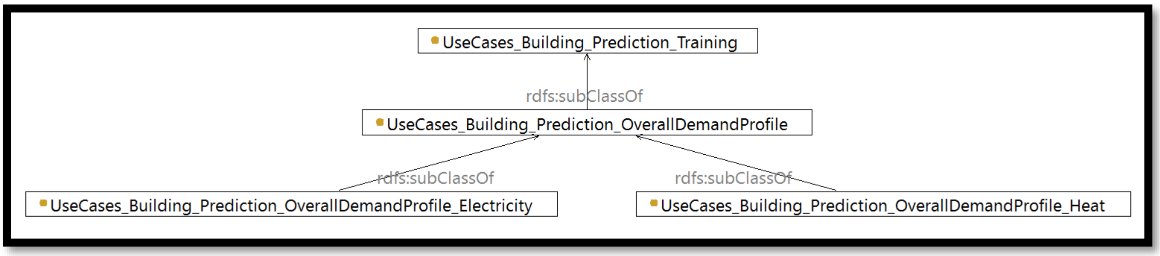



**Figure 14. `UseCases_Building_Prediction_OverallDemandProfile` class and its subclasses.**

Description

This class contains individuals which represent the prediction related use cases applied for a building. The individuals and its properties once reasoned will contain information for developing and training an ANN model which will be applied for day ahead forecasts of overall building heat and electricity demand profiles. The ANN models once trained as a part of this use case, can be consequently used for district optimisation use cases under the class - `UseCases_District_Optimisation_PredictedDemand` class. It has two subclasses as shown above in figure 14.

Relations

- Subclass of `UseCases_Building_Prediction_Training`.
- Disjoint with `UseCases_Building_Prediction_RoomAhu` class and `UseCases_Building_Prediction_SwimmingPoolAhu` class.

Usage of Class

The subclasses of this class are usually defined with individuals rather than this class.

**`UseCases_Building_Prediction_OverallDemandProfile_Heat` Class**

Description

See description of superclass. This class especially looks into prediction of overall heat demand profiles of buildings.

Relations

- Subclass of `UseCases_Building_Prediction_OverallDemandProfile`.
- Disjoint with sibling class `UseCases_Building_Prediction_OverallDemandProfile_Electricity`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of building>*

For example:

*UseCases_Building_Prediction_OverallDemandProfile_Heat_LeisureCentre*

**`UseCases_Building_Prediction_OverallDemandProfile_Electricity` Class**

Similar to class `UseCases_Building_Prediction_OverallDemandProfile_Heat`, but here the focus is on electricity demand profiles prediction of buildings and not heat.

**`UseCases_District_Optimisation_TypicalDemand` class**

Description

This class contains the district optimisation use case but uses a typical demand profile for each building in the district.

Relations

- Subclass of `UseCases_District`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of optimisation related use case for district>*

For example: *UseCases_District_Optimisation_TypicalDemand_District7*

**`UseCases_District_Optimisation_PredictedDemand` class**

Description

This class contains the district optimisation use case but uses predicted demand profile for each building in the district. In other words, the district optimisation uses day ahead demand forecasts for heat and electricity of each building prior to supply side optimisation.

Relations

- Subclass of `UseCases_District`.

Usage of Class

The following naming convention is applied to name individuals of this class:

*<Name of index class>_<Name of optimisation related use case for district>*

For example: *UseCases_District_Optimisation_PredictedDemand_District7*

9. **Classes related to optimisation**

**`Optimisation_ModelParameters_Analytical` class**

Description

The individuals in this class contains all the parameters needed for the analytical model. The individuals are instantiated in the **REMO** ontology. The properties of these individuals are automatically inferred once the ontology is reasoned. These property values, consequently, is used by the analytical model for its calculations.

Relations

- This class is a subclass of `Optimisation_ModelParameters`.
- Disjoint with sibling class: `Optimisation_ModelParameters_Nsga2`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of analytical model parameter>*

For example: *Optimisation_ModelParameters_Analytical_NbOfConsumers*

**`Optimisation_ModelParameters_Nsga2` class**

<u>Description</u>

This class contains individuals which represent NSGA-II algorithm parameters, which is needed for the multiobjective optimisation calculations. The NSGA-II algorithm is used for district schedule optimisation as shown previously. Therefore, the individuals of this class and their properties are predefined based on this study.

<u>Relations</u>

- This class is a subclass of `Optimisation_ModelParameters`.
- Disjoint with sibling class: `Optimisation_ModelParameters_Analytical`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of NSGA-II model parameter>*

For example: *Optimisation_ModelParameters_Analytical_MaximumGenerations*

**`Optimisation_Objectives_ComfortPMV` class**

<u>Description</u>

The individuals in this class represents the comfort factor of Fanger's model called "PMV". This is one of the objectives usually used to monitor indoor comfort in a room within a building. This factor is mainly used as an objective for many building energy management use cases as shown in the SportE2 section.

<u>Relations</u>

- This class is a subclass of `Optimisation_Objectives`.
- Disjoint with sibling classes: `Optimisation_Objectives_OperationalCosts` and `Optimisation_Objectives_OperationalEmissions`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of building>_<Name of Room>*

For example: *Optimisation_Objective_ComfortPMV_LeisureCentre_FitnessRoom*

**`Optimisation_Objective_OperationalEmissions` class**

<u>Description</u>

The individuals in this class contains individuals which represent the 24 hour operational emissions total in the district. This is one of the objectives usually used in the district optimisation model problems. Related to the district energy optimisation use case.

<u>Relations</u>

- This class is a subclass of `Optimisation_Objectives`.
- Disjoint with sibling classes: `Optimisation_Objectives_OperationalCosts` and `Optimisation_Objectives_ComfortPMV`.

<u>Usage of Class</u>

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of district>*

For example: *Optimisation_Objective_OperationalEmissions_District7*

**`Optimisation_Objectives_OperationalCosts` class**

<u>Description</u>

The individuals in this class contains individuals which represent the twent-four hour operational costs total in the district. This is one of the objectives usually used in the district optimisation model problems. Related to the district energy optimisation use case.

<u>Relations</u>

- This class is a subclass of `Optimisation_Objectives`.
- Disjoint with sibling classes: `Optimisation_Objectives_OperationalEmissions` and `Optimisation_Objectives_ComfortPMV`.

Usage of Class

The following naming convention should be applied to any new individuals that needs to be defined in this class:

*<Name of index class>_<Name of district>*

For example: *Optimisation_Objective_OperationalCosts_District7*

## Properties of REMO ontology

### 1. Properties related to `DistrictOperationalParameters` class.

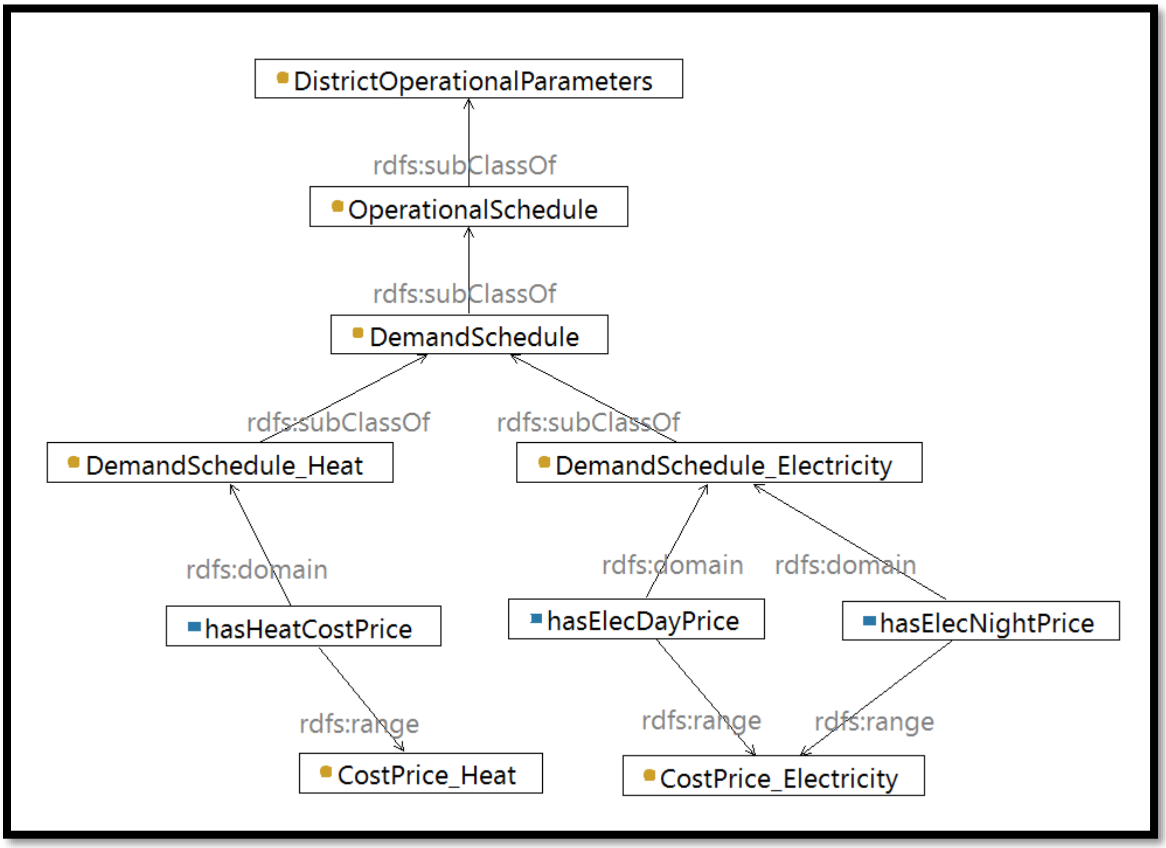



**Figure 15. Properties related to `DistrictOperationalParameters` class and its subclasses.**

**Table 1. Domain and range for properties related to `DistrictOperationalParameters` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasHeatCostPrice*** | `DemandSchedule_Heat` | `CostPrice_Heat` |
| ***hasElecDayPrice*** | `DemandSchedule_Electricity` | `CostPrice_Electricity` |
| ***hasElecNightPrice*** | `DemandSchedule_Electricity` | `CostPrice_Electricity` |

Description

***hasHeatCostPrice*** – this property assigns the cost price of heat energy to the heat demand schedules of the buildings.

***hasElecDayPrice*** - this property assigns the day price of electricity to the electricity demand schedule of the buildings.

***hasElecNightPrice*** – this property assigns the night price of electricity to the electricity demand schedule of the buildings.

2. **Properties related to `EnvironmentalAndFuelProperties` classes and subclasses.**

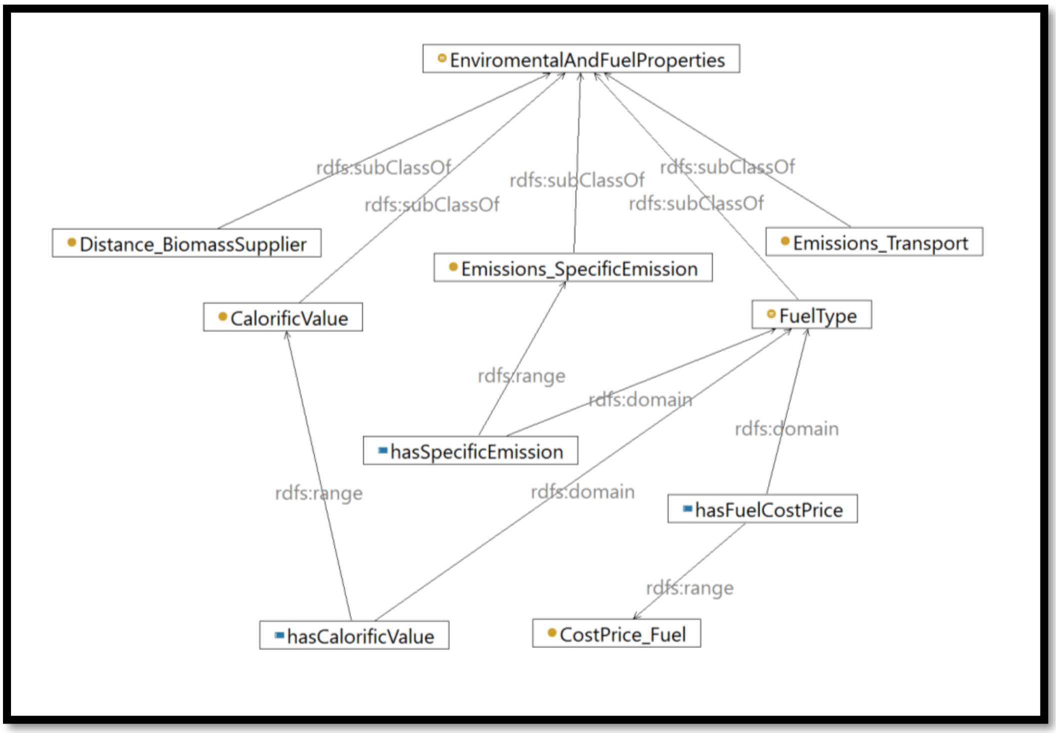



**Figure 16. Properties related to `EnvironmentalAndFuelProperties` class and its subclasses.**

**Table 2. Domain and range for properties related to `EnvironmentalAndFuelProperties` class and its subclasses.**

| Name of Property | Domain | Range |
| --- | --- | --- |
| ***hasSpecificEmission*** | `FuelType` | `Emissions_SpecificEmission` |
| ***hasFuelCostPrice*** | `FuelType` | `CostPrice_Fuel` |
| ***hasCalorificValue*** | `FuelType` | `CalorificValue` |

Description

***hasSpecificEmission*** – this property assigns the specific emission of a particular fuel to its fuel type from the `FuelType` class.

***hasFuelCostPrice***– this property assigns a cost price to the fuel – be it biomass or natural gas or any other type of fuel.

***hasCalorificValue***– this property assigns the calorific value of a fuel to its fuel from the `FuelType` class.

### 3. <u>Properties related to the different subclasses of `EnergySource_Centralsied` class.</u>

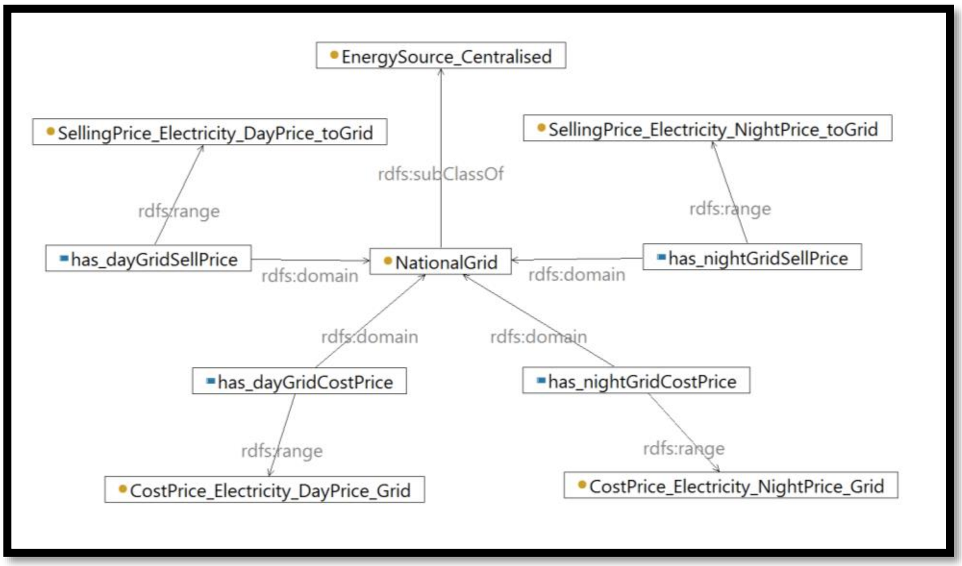



**Figure 17. Properties related to `EnergySource_Centralised` class and its subclasses.**

**Table 3. Domain and range for properties related to `EnergySource_Centralised` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***has_dayGridSellPrice*** | `National Grid` | `SellingPrice_Electricity_DayPrice_toGrid` |
| ***has_nightGridSellPrice*** | `National Grid` | `SellingPrice_Electricity_NightPrice_toGrid` |
| ***has_dayGridCostPrice*** | `National Grid` | `CostPrice_Electricity_DayPrice_fromGrid` |
| ***has_nightGridCostPrice*** | `National Grid` | `CostPrice_Electricity_NightPrice_fromGrid` |

<u>Description</u>

***has_dayGridSellPrice*** – day time selling price of excess electricity production (through decentralised sources) to grid is assigned to a centralised source (`NationalGrid` class) through this property.

***has_nightGridSellPrice*** – night time selling price of excess electricity production to grid is assigned to a centralised source through this property.

***has_dayGridCostPrice*** – the cost price of buying electricity from the grid during day time is assigned here through the property. Once again the domain class is centralised source of energy – `NationalGrid` class.

***has_nightGridCostPrice***– similar to above mentioned property, it assigns the cost price of buying electricity from the grid during night hours.

### 4. Properties related to the different subclasses of `EnergySource_Decentralised` class.

**Table 4. Domain and range for properties related to `EnergySource_Decentralised` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasElecEfficiency*** | EnergySource_ Decentralised _Electricity | Efficiency_ElectricityProduction |
| ***hasHeatEfficiency*** | EnergySource_ Decentralised _Heat | Efficiency_HeatProduction |
| ***hasElec2HeatRatio*** | EnergySource_ Decentralised _Electricity | Elec2heatRatio |
| ***hasIncentive*** | BiomassBoiler | Incentive_BiomassRHI |
| ***HasDistanceToBiomass Supplier*** | BiomassBoiler | Distance_BiomassSupplier |
| ***hasTransportEmission*** | BiomassBoiler | Emissions_Transport |

Description

***hasElecEfficiency*** – this property assigns electricity efficiency to decentralised energy sources.

***hasHeatEfficiency*** – this property assigns heat efficiency to decentralised energy sources.

***hasElec2HeatRatio***– this property assigns the 'electricity to heat ratio' to cogeneration units in the `EnergySource_Decentralised_Electricity` class.

***hasIncentive*** – this property assigns renewable heat incentive to decentralised energy sources such as biomass boiler.

***hasDistanceToBiomassSupplier*** – the distance to biomass fuel supplier from the district is assigned to the biomass boiler energy source class through this property.

***hasTransportEmission*** – this property assigns emissions due to fuel transport (such as biomass fuel transport from supplier to the district) to the biomass boiler energy source class.

## 5. Properties related to `EnergySource` class

**Table 5. Domain and range for properties related to `EnergySource` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasHeatProductionSchedule*** | `EnergySource` | `ProductionScheduleHeat` |
| ***hasFuelType*** | `EnergySource_Decentralised` | `FuelType` |
| ***hasMaxOutputPower*** | `EnergySource_Decentralised` | `OutputPower_Max` |
| ***hasMinOutputPower*** | `EnergySource_Decentralised` | `OutputPower_Min` |
| ***hasMainCost*** | `EnergySource_Decentralised` | `MaintenanceCosts` |

Description

***hasHeatProductionSchedule*** – this property assigns heat production schedules to energy sources.

***hasFuelType*** – this property assigns fuel types to energy sources.

***hasMaxOutputPower*** – this property assigns the maximum output power for an energy source. This is especially needed for district energy optimisation model.

***hasMinOutputPower*** – this property assigns the minimum output power for an energy source. All energy sources can be turned off, but here the minimum output power represents the lowest output power at which an energy source can run without losing much efficiency.

***hasMainCost*** – this property assigns maintenance costs to energy sources – specially to decentralised energy sources.

## 6. Properties related to `EnergyProducerBuilding` class.

**Table 6. Domain and range for properties related to `EnergyProducerBuilding` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***includesElectricitySource*** | EnergyProducerBuilding | EnergySource_Decentralised_Electricity |
| ***includesHeatSource*** | EnergyProducerBuilding | EnergySource_Decentralised_Heat |

Description

***includesElectricitySource*** – this property assigns the electricity sources to energy producer buildings. As they are included in a building, they are decentralised sources and therefore, the range is a subclass of the `EnergySource_Decentralised` class.

***includesHeatSource*** – this property assigns the heat sources to energy producer buildings. The range here is again a subclass of the `EnergySource_Decentralised` class.

## 7. Properties related to `EnergyConsumerBuilding` class.

**Table 7. Domain and range for properties related to `EnergyConsumerBuilding` class and its subclasses.**

| Name of Property | Domain | Range |
|---|---|---|
| ***hasElectricityDemand*** | EnergyConsumer Building | DemandSchedule_Electricity |
| ***hasHeatDemand*** | EnergyConsumer Building | DemandSchedule_Heat |
| ***hasElectricitySource*** | EnergyConsumer Building | NationalGrid |
| | | EnergySource_Decentralised_Electricity |
| ***hasHeatSource*** | EnergyConsumer Building | EnergySource_Decentralised_Heat |

Description

***hasElectricityDemand*** – this property assigns the electricity demand schedule for an energy consumer building.

***hasHeatDemand*** – this property assigns the heat demand schedule for an energy consumer building.

***hasElectricitySource*** – this property assigns the electricity sources which supply electricity to an energy consumer building. It can either be decentralised or centralised sources or sometimes, even both. `NationalGrid` class is a subclass of `EnergySource_Centralised`.

***hasHeatSource*** – this property assigns the heat sources which supply heat to an energy consumer building.

### SPIN rules and constructors

**Rules relevant for overall demand prediction use cases – training and running of prediction models**

For training purposes, once again input and output data is required. The rules below are defined in the class `UseCases_Building_Prediction_OverallDemandProfile_Heat` for inferring the ANN input needed.

**Table 8. Rules used to infer inputs of overall demand prediction models of buildings**

| Rule in SPIN language | Explanation |
|---|---|
| CONSTRUCT {<br>?uc :hasAnnInput ?heatMeter .<br>}<br>WHERE {<br>?uc a :UseCases_Building_Prediction_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?building :hasMainHeatMeter ?heatMeter .<br>} | uc *has Ann Output* heatMeter<br><br>IF uc *belongs to* `UseCases_Building_Prediction_OverallDemandProfile_Heat` class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has main heat meter* heatMeter. |
| CONSTRUCT {<br>?uc :hasAnnInput ?outdoorHum .<br>}<br>WHERE {<br>?uc a :UseCases_Building_Prediction_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?building :hasOutdoorHumSensor ?outdoorHum .<br>} | uc *has Ann Input* outdoorHum<br><br>IF uc *belongs to* `UseCases_Building_Prediction_OverallDemandProfile_Heat` class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has outdoor hum sensor* outdoorHum. |
| CONSTRUCT { | |

| | |
|---|---|
| ?uc :hasAnnInput ?outdoorTemp .<br>}<br>WHERE {<br>?uc a<br>:UseCases_Building_Prediction_OverallDemandProfile_Heat .<br>?uc<br>:isApplicableForTotalDemandPrediction<br>?building .<br>?building :hasOutdoorTempSensor<br>?outdoorTemp .<br>} | uc *has Ann Input* outdoorTemp<br><br>IF uc *belongs to* UseCases_Building_Prediction_OverallDemandProfile_Heat class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has outdoor temp sensor* outdoorTemp. |

The rules infer the main heat and electricity (only the heat meter is shown in the table above) meters of the building, outdoor temperature, and outdoor humidity. The historical data of these meters and sensors is needed for training of the ANN models, and this can be retrieved using SPARQL query after the reasoning process. On the other hand, the output for ANN model training is shown below in Table 9.

**Table 9 . Rules used to infer outputs of overall demand prediction models of buildings**

| Rule in SPIN language | Algorithm |
|---|---|
| CONSTRUCT {<br>?uc :hasAnnOutput ?heatMeter .<br>}<br>WHERE {<br>?uc a<br>:UseCases_Building_Prediction_OverallDemandProfile_Heat .<br>?uc<br>:isApplicableForTotalDemandPrediction<br>?building .<br>?building :hasMainHeatMeter<br>?heatMeter .<br>} | uc *has Ann Output* heatMeter<br><br>IF uc *belongs to* UseCase_Building_Predicion_OverallDemandProfile_Heat class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has main heat meter* heatMeter. |

The output also requires the main heat and electricity meters. Both ANN input and ANN output rely on the same set of historical data for the meters; however, when it comes to the actual selection of data for training, it is different in terms of the timestamp chosen. Similarly, for running these ANN models, the ANN inputs are inferred using rules attached to the class

UseCases_Building_Prediction_Model_OverallDemandProfile_Hea as shown below in Table 10.

**Table 10. Rules used to infer ANN inputs needed for running the overall demand prediction models for buildings**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasAnnInput ?forecastedWeather .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?forecastedWeather a :ForecastedWeatherSchedule_Humidity .<br>} | uc *has Ann Input* forecastedWeather<br><br>IF uc *belongs to* UseCases_Building_Prediction_Model_OverallDemandProfile_Heat class<br><br>AND uc *is applicable for total demand prediction* Building.<br><br>AND forecastedWeather *belongs to* ForecastedWeatherSchedule_Humidity class. |
| **CONSTRUCT** {<br>?uc :hasAnnInput ?forecastedWeather .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?forecastedWeather a :ForecastedWeatherSchedule_Temperature .<br>} | uc *has Ann Input* forecastedWeather<br><br>IF uc *belongs to* UseCases_Building_Prediction_Model_OverallDemandProfile_Heat class<br>AND uc *is applicable for total demand prediction* building.<br>AND forecastedWeather *belongs to* ForecastedWeatherSchedule_Temperature class. |
| **CONSTRUCT** {<br>?uc :hasAnnInput ?heatMeter .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?building :hasMainHeatMeter ?heatMeter.<br>} | uc *has Ann Input* heatMeter.<br><br>IF uc *belongs to* UseCases_Building_Prediction_Model_OverallDemandProfile_Heat class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has main heat meter* heatMeter. |

For running of ANN models, individuals from the `ForecastedWeatherSchedule` class, which is a subclass of `DistrictOperationalParameters`, are inferred as input along with the meter readings. To further run these ANN models, SPARQL query is used to retrieve real-time information from BMS/EMS and it is then post-processed as per the ANN model requirements.

Similarly, ANN outputs are inferred using the rules as shown in Table 11 below:

**Table 11. Rules used for ANN outputs while running the overall demand prediction of buildings scenario**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasAnnOutput ?predictedDemand .<br>}<br>**WHERE** {<br>?uc a :UseCases_Building_Prediction_Model_OverallDemandProfile_Heat .<br>?uc :isApplicableForTotalDemandPrediction ?building .<br>?building :hasPredictedHeatDemand ?predictedDemand .<br>} | uc *has Ann Output* predictedDemand.<br><br>IF uc *belongs to* `UseCases_Building_Prediction_Model_OverallDemandProfile_Heat` class<br>AND uc *is applicable for total demand prediction* building.<br>AND building *has predicted heat demand* predictedDemand. |

The individuals inferred as ANN output are further queried to retrieve the location for recording the predicted demand. The SPARQL query used here is shown in the validation Section, 7.1.3.

For electricity demand profile prediction of the overall building, the same rules are applied to the classes `UseCases_Building_Prediction_OverallDemandProfile_Electricity` and `UseCases_Building_Prediction_Model_OverallDemandProfile_Electricity`, with the exception that the building's electricity meter is used instead of the heat meter.

**Rules relevant for use cases representing the district optimisation using typical demand**

This set of rules is attached to the class `UseCases_District_Optimisation_TypicalDemand`.

Optimisation settings are inferred mainly from the class `Optimisation_ModelParameters_Nsga2`. All individuals under this class are inferred through reasoning and assigned to the respective individuals of the `UseCases_District_Optimisation_TypicalDemand` class by using the *hasOptimSettings* property. The rules are shown below in Table 12.

**Table 12. Rules used for inferencing the district optimisation-related parameters**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasOptimSettings ?settings .<br>}<br>**WHERE** {<br>?uc a<br>:UseCases_District_Optimisation_TypicalDeman<br>d .<br>?settings a<br>:Optimisation_ModelParameters_Nsga2 .<br>} | uc *has optim settings* settings<br><br>IF uc *belongs to* `UseCases_District_Optimisation_TypicalDemand` class<br><br>AND settings *belongs to* `Optimisation_ModelParameters_Nsga2` class |

The *hasOptimModelParameters* property is also very similar to the *hasOptimSettings* property. Once again, a rule is defined (shown in Table 13) through which all the additional parameters and information needed for running the district analytical model are assigned to the respective individuals of the `UseCases_District_Optimisation_TypicalDemand` class through the *hasOptimModelParameters* property. These parameters are default instances defined under the class `Optimisation_ModelParameters_Analytical`.

**Table 13. Rules used for inferencing the district analytical model-related parameters**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?uc :hasOptimModelParameters ?parameters<br>.<br>}<br>**WHERE** { | uc *has optim settings* parameters<br><br>IF uc *belongs to* `UseCases_District_Optimisation_TypicalDemand` class |

| | |
|---|---|
| ?uc a :UseCases_District_Optimisation_TypicalDemand . ?parameters a :Optimisation_ModelParameters_Analytical . } | AND parameters *belongs to* Optimisation_ModelParameters_Analytical class |

The optimisation objectives here are inferred from the instances under the classes Optimisation_Objectives_OperationalCosts and Optimisation_Objectives_OperationalEmissions, which are the two objectives of district optimisation problems. They are inferred using the rules shown below in Table 14.

**Table 14. Rules used to infer the objectives of the district optimisation use cases**

| Rule in SPIN language | Algorithm |
|---|---|
| **CONSTRUCT** {<br>?uc :hasOptimObjective ?obj1 .<br>?uc :hasOptimObjective ?obj2 .<br>}<br>**WHERE** {<br>?uc a :UseCases_District_Optimisation_TypicalDemand .<br>?obj1 a :Optimisation_Objectives_OperationalCosts .<br>?obj2 a :Optimisation_Objectives_OperationalEmissions .<br>} | uc *has optim objective* obj1<br>AND uc *has optim objective* obj2<br><br>IF uc *belongs to* UseCases_District_Optimisation_TypicalDemand class<br><br>AND obj1 *belongs to* Optimisation_Objectives_OperationalCosts class<br><br>AND obj2 *belongs to* Optimisation_Objectives_OperationalEmissions class. |

The decision variables of the district optimisation use case are usually the production schedules of the energy sources in the energy producer building. The energy producer building here is the target to which the use case is applied to in the first place – which is defined by the user during the instantiation process. Expressing the above-mentioned relationships as rules can infer the decision variables of the optimisation problem as shown below in Table 15.

**Table 15. Rules used to infer the decision variables of the district optimisation use cases.**

| Rule in SPIN language | Algorithm |
|---|---|
| CONSTRUCT {<br>    ?uc :hasDecisionVariable ?dv .<br>}<br>WHERE {<br>    ?uc a :UseCases_District_Optimisation_TypicalDemand .<br>    ?uc :isApplicableForDistrictOptimisation ?producerBuilding .<br>    ?producerBuilding :includesHeatSource ?source .<br>    ?source :hasHeatProductionSchedule ?dv .<br>} | uc *has Decision Variable* dv<br><br>IF uc *belongs to* `UseCases_District_Optimisation_TypicalDemand` class<br><br>AND uc *is applicable for district optimisation* producerBuilding<br><br>AND producerBuilding *includes heat source* source<br><br>AND source *has heat production schedule* dv |

The example above refers to heat production schedule optimisation problems only. Similar work can be carried out for electricity production schedules.

**Rules relevant to use cases representing district optimisation using predicted demand**

The rules here are similar to the section above; the only difference is that here the use case individual belongs to the subclass `UseCases_District_Optimisation_PredictedDemand`.

More details on the actual results of running these rules post-reasoning of the ontology are shown in Section 7.1.2.

**Rules used for inferring numerical values**

Similarly, the number of energy sources in the district can also be inferred as shown below in Table 16.

**Table 16. Rules used to infer a numerical value for the number of energy sources in the district.**

| Rule in SPIN language | Explanation |
|---|---|
| CONSTRUCT {<br>    ?individual :hasAnalyticalModelValue ?count .<br>}<br>WHERE {<br>    { | Individual *has Analytical model value* count<br><br>WHERE<br>{<br>    DISTINCT Number of sources is count<br>    AND sources *belong to* cl<br>    AND cl belongs to subclass of `EnergySource_Decentralised_Heat` class |

| | |
|---|---|
| SELECT ((COUNT(DISTINCT<br>?sources)) AS ?count)<br>WHERE {<br>?sources a ?cl .<br>?cl rdfs:subClassOf<br>:EnergySource_Decentralised_Heat .<br>}<br>} .<br>?individual a<br>:Optimisation_ModelParameters_Analytical .<br>FILTER regex(str(?individual),<br>"NbOfGenerationUnits") .<br>} | }<br>AND individual *belongs to* `Optimisation_ModelParameters_Analytical` class<br><br>FILTER individual *with name* "NbOfGenerationUnits" |

Another example is to infer the distance between the biomass supplier and the district, which can be inferred as shown below in Table 17.

**Table 17. Rules used to infer distance between biomass supplier and the district.**

| Rule in SPIN language | Explanation |
|---|---|
| **CONSTRUCT** {<br>?individual :hasAnalyticalModelValue<br>?distance.<br>}<br>**WHERE** {<br>?individual a<br>:Optimisation_ModelParameters_Analytical .<br>**FILTER** regex(str(?individual), "Distance") .<br>{<br>**SELECT** ?distance<br>**WHERE** {<br>?value :hasNumericalValue ?distance .<br>?value a :ScalarValueClass .<br>?dist :hasValueName ?value .<br>?dist a :Distance_BiomassSupplier .<br>}<br>} .<br>} | Individual *has Analytical model value* distance<br><br>IF individual *belongs to* `Optimisation_ModelParameters_Analytical` class<br><br>FILTER individual *with name* "Distance"<br><br>AND<br>{SELECT Distance<br>WHERE<br>{Value *has numerical value* distance<br>AND<br>Value *belongs to* `ScalarValueClass`<br>AND<br>Dist *has value name* value<br>AND<br>Dist *belongs to* `Distance_BiomassSupplier`<br>}<br>} |

Similarly, values for transport emissions and renewable heat incentives can also be derived from rules. These are some of the parameters which are inferred by SPIN rules and used by the district optimisation model. Other relevant parameters can be queried through SPARQL, as explained in the validation and instantiation section (Section 7.1.3).

*Constructors*

### Constructors for creating properties for energy sources

Some of the energy sources, when created, would need to have default properties instantiated as well. Using constructors here again makes the instantiation process semi-automated, as the user does not need to create instances for these properties. Table 18 below shows constructors defined in the `BiomassBoiler` class. As a result of this, during the instantiation process, when the user defines an individual instance of biomass boiler, an instance of each property of biomass boiler is defined (relevant to **REMO** ontology) under its respective class.

**Table 18. Constructors defined for `BiomassBoiler` class**

| Rule in SPIN language | Explanation |
|---|---|
| CONSTRUCT {<br>?this :hasHeatEfficiency ?heatEff .<br>?heatEff a :Efficiency_HeatProduction .<br>}<br>WHERE {<br>?this a :BiomassBoiler .<br>BIND (str(?this) AS ?x) .<br>BIND (STRAFTER(?x, "#") AS ?y) .<br>BIND (STRBEFORE(?x, "#") AS ?uri)<br>BIND (URI(CONCAT(?uri,"#Efficiency_heatProduction_", ?y)) AS ? heatEff) .<br>} | this *has value name* heatEff<br><br>AND heatEff *belongs to* `Efficiency_HeatProduction`<br><br>IF this *belongs to* `BiomassBoiler` class<br><br>AND BIND (string value of variable (this)) *AS* x<br><br>AND BIND (string which comes after “#” in x) AS y<br><br>AND BIND (string which comes before “#” in x) AS uri<br><br>AND BIND (concatenate strings: uri, "#Efficiency_heatProduction_", y ) AS heatEff |
| CONSTRUCT {<br>?this :hasHeatProductionSchedule ?sch .<br>?sch a :ProductionScheduleHeat_Typical .<br>}<br>WHERE {<br>?this a :BiomassBoiler .<br>BIND (str(?this) AS ?x) .<br>BIND (STRAFTER(?x, "#") AS ?y) .<br>BIND (STRBEFORE(?x, "#") AS ?uri) | this *has heat production schedule* sch<br>AND sch *belongs to* `ProductionScheduleHeat_Typical`<br><br>IF this *belongs to* `BiomassBoiler` class<br>AND BIND (string value of variable (this)) *AS* x<br><br>AND BIND (string which comes after “#” in x) AS y |

| | |
|---|---|
| BIND<br>(URI(CONCAT(?uri,"#ProductionSchedule_", ?y))<br>AS ?sch) .<br>} | AND BIND (string which comes before "#" in x) AS uri<br><br>AND BIND (concatenate strings: uri, "#ProductionSchedule_", y ) AS sch |
| CONSTRUCT {<br>?this :hasMainCost ?mainCost.<br>? mainCost a :MaintenanceCosts .<br>}<br>WHERE {<br>?this a :BiomassBoiler .<br>BIND (str(?this) AS ?x) .<br>BIND (STRAFTER(?x, "#") AS ?y) .<br>BIND (STRBEFORE(?x, "#") AS ?uri)<br>BIND<br>(URI(CONCAT(?uri,"#MaintenanceCosts_", ?y))<br>AS ? mainCost) .<br>} | this *has Main Cost* mainCost<br>AND mainCost *belongs to* `ProductionScheduleHeat_Typical`<br><br>IF this *belongs to* `BiomassBoiler` class<br>AND BIND ( string value of variable (this)) *AS* x<br><br>AND BIND (string which comes after "#" in x) AS y<br><br>AND BIND (string which comes before "#" in x) AS uri<br><br>AND BIND (concatenate strings: uri, "#MaintenanceCosts_", y) AS mainCost |
| CONSTRUCT {<br>?this :hasMaxOutputPower ?maxPower .<br>?maxPower a :OutputPower_Max .<br>}<br>WHERE {<br>?this a :BiomassBoiler .<br>BIND (str(?this) AS ?x) .<br>BIND (STRAFTER(?x, "#") AS ?y) .<br>BIND (STRBEFORE(?x, "#") AS ?uri)<br>BIND<br>(URI(CONCAT(?uri,"#OutputPower_Max_", ?y))<br>AS ?maxPower) .<br>} | this *has heat Main Cost* maxPower<br>maxPower *belongs to* `OutputPower_Max`<br><br>IF this *belongs to* `BiomassBoiler` class<br><br>AND BIND (string value of variable (this)) *AS* x<br><br>AND BIND (string which comes after "#" in x) AS y<br><br>AND BIND (string which comes before "#" in x) AS uri<br><br>AND BIND (concatenate strings: uri, "#OutputPower_Max_", y) AS maxPower |

Similarly, some of the other properties for biomass boiler individuals such as minimum output power, fuel type, incentives, distance to biomass supplier and so forth can also be instantiated using constructors. Likewise, individuals of the `CombinedHeatPower` class and `GasBoiler` class also have default properties defined.

**Constructors for creating BMS and EMS locations for each dynamic parameter**

This constructor creates and assigns an instance under the `ParameterMapping_BMS` or `ParameterMapping_EMS` class for every individual created representing a dynamic parameter in **REMO** ontology (such as sensors, actuators, production schedules, demand schedules, meter readings and so forth). The constructor is attached to the class that represents the dynamic parameter. For example, Table 19 below shows the constructor applied to individuals of the `Actuators` class defined by the user. The constructor here creates and assigns an individual instance for the newly defined actuator representing its location in BMS.

**Table 19. Constructors defined for `ParameterMapping_BMS` class**

| Rule in SPIN language | Explanation |
|---|---|
| CONSTRUCT {<br>?this :has_locationBMS ?new .<br>?new a :ParameterMapping_BMS .<br>}<br>WHERE {<br>?this a :Actuators .<br>BIND (str(?this) AS ?x) .<br>BIND (STRAFTER(?x, "#") AS ?y) .<br>BIND (STRBEFORE(?x, "#") AS ?uri)<br>BIND (URI(CONCAT(?uri,"#ActuatorLocation_", ?y)) AS ?new) .<br>} | this *has location in BMS* new<br>new *belongs to* `ParameterMapping_BMS` class<br><br>IF this *belongs to* `Actuators` class<br>AND BIND (string value of variable (this)) *AS* x<br>AND BIND (string which comes after “#” in x) AS y<br>AND BIND (string which comes before “#” in x) AS uri<br>AND BIND (concatenate strings: uri, "#ActuatorLocation_", y) AS new |

Following this, the user simply needs to assign a string value to the individual created using the relevant properties. The string value here represents the actual location endpoint defined in the BMS through which the real-time value can be accessed. In the case of the actuators, they also have to be assigned string value, which represents the location to modify the setpoints in the BMS.

# Appendix B – MATLAB code for analytical model and its optimisation

## Running the analytical model

**eedistrict.m** is the main Matlab file which when executed runs the analytical model. This file calls various other functions and files within the code. Figure 18 below represents the workflow in which the various other files and functions are called within eedistrict.m.

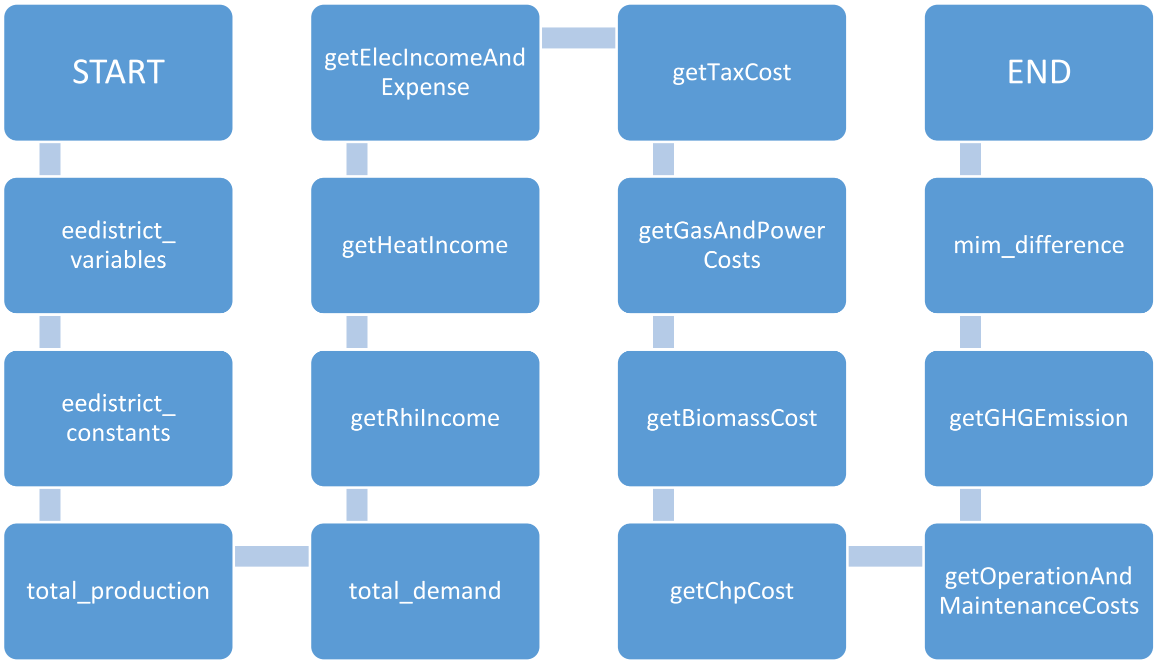



**Figure 18. Workflow of files running in eedistrict.m**

Below the code of **eedistrict.m** is shown.

```
% "eedistrict.m" is the main file which calls all the separate functions for the analytical model to run.

%*******************************************************************

%{ the .m files below initialises all the constants and the variables needed for the analytical model to run %}

eedistrict_variables;

eedistrict_constants;

%*****************************************************************

% {call function "total_production" below to calculate the following: the electricity production, the daily total heat energy produced, the daily total electricity energy produced %}

[dailytotal_heatProductionArray, electricityProductionSchedule,
dailytotal_electricityProductionArray ]=total_production
(heatProductionSchedule,ElectricityToHeatRatios,NbOfGenerationUnits,NbOfTimeslots);

%*****************************************************************

% reading 24-hour demand data for each building using "xlsread" function.

heatDemand = xlsread('heat_demand.xls');

electricityDemand = xlsread('electricity_demand.xls');

%*****************************************************************
```

```
% “total_demand” function is called to calculate the daily total heat energy and electricity energy demand %}

[totalHeatDemand,totalElectricityDemand]=total_demand(heatDemand,electricityDemand);

%*******************************************************************

%{“getRhiIncome” function is called to calculate the income from Renewable Heat Incentive (RHI) for using the biomass boilers %}

rhi_income= getRhiIncome(BiomassIndexes,BiomassRHI,dailytotal_heatProductionArray);

% “getHeatIncome” function is called to calculate the income from selling heat to the learning zone.

 heat_income = getHeatIncome(dailytotal_heatProductionArray,HeatSalePriceToConsumer);

%{ electricity income and expense is calculated by function “getElecIncomeAndExpense” which is called below %}

[elec_income,elec_expense] =
getElecIncomeAndExpense(DayLimit,NbOfTimeslots,electricityDemand,NbOfGenerationUnits,electricityProductionSchedule,ElectricityDayPurchaseRateFromGrid,ElectricityDaySalePriceToConsumer,ElectricityNightPurchaseRateFromGrid,ElectricityNightSaleRateToGrid );

%carbon taxes are calculated by calling function “getTaxCost” as shown below

taxCost =
getTaxCost(dailytotal_heatProductionArray,GasFueledSourceIndexes,GenerationUnitEfficiencies,CRCNaturalGasConversionRate,CRCTaxRate);

%{ costs of fuel for CHP, Biomass, boilers are calculated by calling the function
“getGasAndPowerCosts”, “getBiomassCost”,and  “getChpCost” %}

gasCost =
getGasAndPowerCosts(dailytotal_heatProductionArray,GasBoilerIndexes,GenerationUnitEfficiencies,NaturalGasPurchasePrice);

biomassCost =
getBiomassCost(dailytotal_heatProductionArray,BiomassIndexes,BiomassPurchasePrice);

ChpCost =
getChpCost(dailytotal_heatProductionArray,ChpIndexes,NaturalGasPurchasePrice,GenerationUnitEfficiencies);

%{ costs for operations & maintenance of CHP can be calculated by calling function
“getOperationAndMaintenanceCosts” as shown below.%}

omCost=
getOperationAndMaintenanceCosts(ChpIndexes,dailytotal_electricityProductionArray,ChpMaintenanceRate);

%{ ‘objectives’ variable below represents both cost and emissions. ‘objectives(2)’ represents cost and ‘objectives(1)’ represents emissions. %}
```

```
objectives(2)= (heat_income+rhi_income+elec_income-elec_expense-taxCost-gasCost-biomassCost-ChpCost-omCost);

%****************************************************************************

% greenhouse gas emissions are computed using the  "getGHGEmission" function

objectives(1)=getGHGEmission(heatProductionSchedule,NbOfGenerationUnits, NbOfTimeslots,SpecificGasEmissions,BiomassIndexes,dailytotal_heatProductionArray,Btrc,BiomassCalorificValue,GenerationUnitEfficiencies,DistanceToBiomassSupplier);

%*********************************************************************

%function to calculate the difference between production and demand. This will act as constraints to the optimisation problem.

[x,co]=
min_difference(totalHeatDemand,dailytotal_heatProductionArray,heatProductionSchedule,heatDemand);

%*********************************************************************
```

**eedistrict_constants.m**

```
% this file defines all the constants that are going to be used in the energy calculations

NbOfGenerationUnits=7;

NbOfConsumers=5;

NbOfTimeslots=48;

TransportEmissions=0.0001231;

SpecificGasEmissions=[0.185 0.015 0.015 0.185 0.185 0.185 0.185];

ElectricityToHeatRatios= [0.65 0 0 0 0 0 0];

GenerationUnitEfficiencies=[0.78 0.82 0.82 0.67 0.67 0.67 0.67];

GenerationUnitLowerBounds_Kwh= [187.5 62 62 0 0 0 0];

GenerationUnitUpperBounds_Kwh= [200.5 247.5 247.5 800 800 800 800];

nboftypes=3;

productionType= [1 0 0 0;2 3 0 0; 4 5 6 7];

BiomassIndexes= [2 3];

ChpIndexes=1;

GasBoilerIndexes=[4 5 6 7];

GasFueledSourceIndexes=[1 4 5 6 7];

NbOfBiomass=2;
```

```
NbOfChp=1;
NbOfGasBoilers=4;
BiomassUpperBound=124;
ChpUpperBound=401;
GasBoilerUpperBound=800;
ElectricityNightSaleRateToGrid=0.03;
ElectricityDaySaleRateToGrid=0.0764;
ElectricityNightPurchaseRateFromGrid=0.07;
ElectricityDayPurchaseRateFromGrid=0.11;
ElectricityNightSalePriceToConsumer=0.07;
ElectricityDaySalePriceToConsumer=0.11;
NaturalGasPurchasePrice=0.0248;
NaturalGasCalorificValue=10.56;
BiomassPurchasePrice=0.205;
BiomassCalorificValue=4.8;
DistanceToBiomassSupplier=277;
HeatSalePriceToConsumer=0.0594;
CRCTaxRate=12;
CRCElectricityConversionRate=0.541;
CRCNaturalGasConversionRate=0.1836;
ChpMaintenanceRate=0.0035;
BiomassRHI=0.12;
Btrc=0.00012;
SupplySafetyMargin=0.2;
DayLimit=14;
%**********************************************************
```

**eedistrict_variables.m**

```
%this file initialises the variables which are needed with default values.
heatDemand=[0:47 ; 0:47 ; 0:47 ; 0:47; 0:47];
totalHeatDemand =[0:5];
electricityDemand=[0:47 ; 0:47 ; 0:47 ; 0:47; 0:47 ; 0:47];
```

```
totalElectricityDemand=[0:6];

heatProductionSchedule=xlsread('heat_production_schedule.xls');

objectives = [100,100]; % cost, emissions % this is the output of the cost model

dailytotal_heatProductionArray =[0:7];

dailytotal_electricityProductionArray =[0:7];

%************************************************************
```

**total_production.m**

```
%{ this is the function to generate total heat and total electricity production for each time slot of the day %}

function [dailytotal_hpArray,electricityPs,dailytotal_epArray ]=
total_production(heatProductionSchedule,ElectricityToHeatRatios,NbOfGenerationUnits,NbOf
Timeslots)

dailytotal_hpArray = sum(heatProductionSchedule,2);

dailytotal_hpArray =transpose(dailytotal_hpArray );

for i=1: NbOfGenerationUnits

            for t=1: NbOfTimeslots

        electricityPs(i,t)=ElectricityToHeatRatios(i)*heatProductionSchedule(i,t);

   end

end

dailytotal_epArray =sum(electricityPs,2);

dailytotal_epArray=transpose(dailytotal_epArray);

end

%*******************************************************************
```

**total_demand.m**

```
% this is the function to calculate the total heat and total electricity demand for the day

function [dailytotal_hdArray,dailytotal_edArray ]=
total_demand(heatDemand,electricityDemand)

dailytotal_hdArray = sum(heatDemand,2);

dailytotal_hdArray =transpose(dailytotal_hdArray);

dailytotal_hdArray = dailytotal_hdArray *0.2 +dailytotal_hdArray; % this is to add the 20 % loss factor

dailytotal_edArray =sum(electricityDemand,2);
```

```
dailytotal_edArray=transpose(dailytotal_edArray);
end
%****************************************************************
```

**getRhiIncome.m**

```
function [rhi_income] = getRhiIncome( biomass_index,biomass_rhi,daily_total_heat_produc)
%Calculate the amount of incentive received for using biomass boilers
rhi_income= 0;
for i=1:size(biomass_index,2)
   rhi_income = rhi_income + (daily_total_heat_produc(biomass_index(i)) * biomass_rhi);
end
end
%****************************************************************
```

**getHeatIncome.m**

```
% function to calculate the income due to selling of heat energy.
function [heat_income] = getHeatIncome( daily_total_heat,HeatSalePriceToConsumer)
% only the learningZone is charged for heating as rest of the buildings are owned by the council.
   daily_total_heat=daily_total_heat-0.2*daily_total_heat;
   g=1; % heating demand of the learning zone is stored in index 1
         heat_income =  HeatSalePriceToConsumer * daily_total_heat(g);
end
%*****************************************************
```

**getElecIncomeAndExpense.m**

```
%function to calculate the electricity income and expense in the district
function [income,expense] = getElecIncomeAndExpense(daylimit,nb_time_slots,elec_demand,Nb_generation,elec_produc,day_purchase_grid,day_sell_grid,night_purchase_grid,night_sell_grid )
%This function calculates income from electricity and expense of
%electricity used by the district
learningDayElectricityConsumption = 0;
ehubDayElectricityConsumption = 0;
day_expense=0;
```

```
day_income=0;

%DAY

for i=daylimit:nb_time_slots

   learningDayElectricityConsumption = learningDayElectricityConsumption+ elec_demand(1,i);

   ehubDayElectricityConsumption = ehubDayElectricityConsumption +elec_demand(2,i);

end

dayElectricityProduction = 0;

for j=1:Nb_generation

   for k=daylimit:nb_time_slots

      dayElectricityProduction=dayElectricityProduction + elec_produc(j,k);

   end

end

if (dayElectricityProduction < ehubDayElectricityConsumption)

   elec_bought_from_grid = ehubDayElectricityConsumption-dayElectricityProduction;

   day_expense=elec_bought_from_grid*day_purchase_grid;

else
if(dayElectricityProduction>(ehubDayElectricityConsumption+learningDayElectricityConsumpti
on))

      elec_sold_to_grid = (dayElectricityProduction-
(ehubDayElectricityConsumption+learningDayElectricityConsumption));

      day_income=elec_sold_to_grid*day_sell_grid;

      elec_sold_to_consumer=learningDayElectricityConsumption;

      day_income = day_income+elec_sold_to_consumer*day_purchase_grid;

   else

      elec_sold_to_consumer=dayElectricityProduction-ehubDayElectricityConsumption;

      day_income=elec_sold_to_consumer*day_purchase_grid;

   end

end

%NIGHT

   learningNightElectricityConsumption = 0;

   ehubNightElectricityConsumption = 0;

   night_expense=0;
```

```
  night_income=0;
 for l=1:(daylimit-1)
     learningNightElectricityConsumption = learningNightElectricityConsumption+
elec_demand(1,j);
     ehubNightElectricityConsumption = ehubNightElectricityConsumption +elec_demand(2,j);
  end
 NightElectricityProduction = 0;
 for m=1:Nb_generation
     for n=1:(daylimit-1)
        NightElectricityProduction=NightElectricityProduction + elec_produc(m,n);
     end
 end
 if (NightElectricityProduction < ehubNightElectricityConsumption)
     night_elec_bought_from_grid = ehubNightElectricityConsumption-
NightElectricityProduction;
     night_expense=night_elec_bought_from_grid*night_purchase_grid;


  else
if(NightElectricityProduction>(ehubNightElectricityConsumption+learningNightElectricityConsu
mption))
        night_elec_sold_to_grid = NightElectricityProduction-
(ehubNightElectricityConsumption+learningNightElectricityConsumption);
        night_income=night_elec_sold_to_grid*night_sell_grid;
        night_elec_sold_to_consumer=learningNightElectricityConsumption;
        night_income=night_income+ night_elec_sold_to_consumer*night_purchase_grid;
     else
        night_elec_sold_to_consumer=NightElectricityProduction-
ehubNightElectricityConsumption;
        night_income=night_elec_sold_to_consumer*night_purchase_grid;
     end
  end
 income =day_income+night_income;
 expense=day_expense+night_expense;
```

```
end
%*************************************************************************
```

**getTaxCost.m**

```
% function to calculate the tax incurred due to carbon production
function [tax_cost] = getTaxCost(dailytotal_heatProduction,
gas_fuel_source_index,genunit_efficiency,CRC_Naturalgas_conversionRate,CRC_taxrate)
   tax_cost=0;
   for g= 1:size(gas_fuel_source_index,2)
      taxForGenUnit = (dailytotal_heatProduction(gas_fuel_source_index(g)) /
genunit_efficiency(gas_fuel_source_index(g))) * (CRC_Naturalgas_conversionRate) / 1000*
CRC_taxrate;
      tax_cost = tax_cost + taxForGenUnit;
   end
end
%*************************************************************************
```

**getGasAndPowerCosts.m**

```
% fucntion to calculate the cost of fuel(gas,biomass) and power incurred
function
[gasCost]=getGasAndPowerCosts(daily_total_heat_Production,gas_boiler_ind,gen_unit_eff,Ng
_price)
% calculating gas cost of gas boilers
gasCost=0;
for g=1:(size(gas_boiler_ind,2))
   gasConsumption = daily_total_heat_Production(gas_boiler_ind(g)) /
gen_unit_eff(gas_boiler_ind(g));
   gasCost = gasCost + Ng_price * gasConsumption;
   end
end
%*************************************************************************
```

**getChpCost.m**

```
% this function is used calculate the cost of fuel used by CHP
function [ChpCost]=getChpCost(daily_total_heat_Production,chp_ind,gasPrice,gen_unit_eff)
   ChpCost=0;
```

```
   for g=1:size(chp_ind,2)

   ChpCost = ChpCost+ gasPrice*(daily_total_heat_Production(chp_ind(g))/
gen_unit_eff(chp_ind(g)));

   end

end

%*********************************************************************
```

**getBiomassCost.m**

```
% this function is used calculate the cost of biomass fuel used by biomass boilers

function [bCost]=getBiomassCost(daily_total_heat_Production,biomass_ind,biomassPrice)

      % calculating the biomass fuel cost

      bCost=0;

      for i=1:(size(biomass_ind,2))

         bCost = bCost + ( biomassPrice* (1 / 4.7)* daily_total_heat_Production(biomass_ind(i)));

         % include the 1/4.8 factor in the parameters inverse of the net

         %calorific value of biomass pellets

      end

   end

%**********************************************************************
```

**getOperationAndMaintenanceCosts.m**

```
% this function calculates the maintenance cost of the CHP

function [omCost]=
getOperationAndMaintenanceCosts(Chp_indixes,dailytotal_elec_Production,
Chp_maintenance_rate)

 omCost=0;

for g=1:size(Chp_indixes,2)

 maintenanceForGenUnit =
dailytotal_elec_Production(Chp_indixes(g))*Chp_maintenance_rate;

    omCost = omCost + maintenanceForGenUnit;

end

end

%***********************************************************************
```

**getGHGEmission.m**

```
%{ this function is used to calculate the total green house gas emissions from the production in the district. %}

function [emissions]=getGHGEmission(heatProductionSch,no_generation_units,
no_of_timeslots,specific_emissions,
biomass_ind,daily_heatProductionArray,btrc,Bio_Calorific,gUnitEfficiencies,distToBiomassSupplier)

emissions = 0;

for i=1:no_generation_units

   for j=1:no_of_timeslots

   emissions = emissions + (heatProductionSch(i,j)*specific_emissions(i));

   end

end

 %transport emissions

for k=1:(size(biomass_ind,2))

x= btrc * distToBiomassSupplier * daily_heatProductionArray(biomass_ind(k));

y= Bio_Calorific * gUnitEfficiencies(biomass_ind(k));

emissions = emissions + (x/y)  ;

end

%**********************************************************************
```

**Min_difference.m**

```
function [ min_dif, cons] =
min_difference(totalHeatDemand,dailytotal_heatProductionArray,heatProduction,heatDemand)

%this fucntion returns the difference between the total heat produced and total heat demand.

min_dif = abs(sum(totalHeatDemand)-sum(dailytotal_heatProductionArray));

hProduction = sum(heatProduction,1);

hDemand = sum(heatDemand,1);

for i=1:48

   c(i)= hProduction(i)-hDemand(i);

   disp('hProduction(i)')

    disp(hProduction(i))

   disp('hDemand(i)')

   disp(hDemand(i))
```

```
  disp('c(i)');
  disp(c(i));
  if (c(i)>=-1)
     cons(i)=0;
    disp(cons(i))
  else
     cons(i)=abs(c(i));
     disp('no meet')
     disp(cons(i));
  end
end
```

## Optimisation using NSGA-II, and analytical model as cost function

The following code was applied to run optimisation.

```
opt = nsgaopt_modified(); % sets the options for the NSGA-II
nsga2(opt); % calls the nsga2 function to start the optimisation process. Nsga2 file is adopted from (reference…)
```

Optimisation was run using files from (Aravind Seshadri, 2006)[22]

**Nsgaopt_modified.m**

```
%this file assigns all the settings for the NSGA-II optimisation code.
function defaultopt = nsgaopt_modified()
order={'chp'; 'biomass'; 'gas'};
if (strcmp(order(1),'chp'))
    for b=1:7
    for c=1:48
      if (b==1)
         lb(b,c)=187.5;
      elseif (b==2)||(b==3)
         lb(b,c)=62;
      else
```

[22] https://uk.mathworks.com/matlabcentral/fileexchange/10429-nsga-ii--a-multi-objective-optimization-algorithm

```
                lb(b,c)=0;
            end
        end
        end
 end
lb_tran=lb';
   lb= reshape(lb_tran,1,[]);
if (strcmp(order(1),'chp'))
        for b=1:7
        for c=1:48
            if (b==1)
                ub(b,c)=201;
            elseif (b==2)||(b==3)
                ub(b,c)=247.5;
            else
                ub(b,c)=800;
            end
        end
        end
 end
ub_tran=ub';
   ub= reshape(ub_tran,1,[]);
defaultopt = struct(...
... % Optimization model
   'popsize',20,...          % population size
   'maxGen', 10,...          % maximum generation
   'numVar', 336,...          % number of design variables
   'numObj', 2,...          % number of objectives
   'numCons', 48,...          % number of constraints
   'lb',lb,...              % lower bound of design variables [1:numVar]
   'ub', ub,...              % upper bound of design variables [1:numVar]
   'vartype', [],...          % variable data type [1:numVar]1=real, 2=integer
```

```
   'objfun', @eedistrict_function,...      % objective function
... % Optimization model components' name
   'nameObj',{{}},...
   'nameVar',{{}},...
   'nameCons',{{}},...
... % Initialization and output
...% ,oldresult,ngen} add this if you need initial population
   'initfun', {{@initpop}},...       % population initialization function (use random number as default)
   'outputfuns',{{@output2file}},...   % output function
   'outputfile', 'populations.txt',... % output file name
   'outputInterval', 1,...           % interval of output
   'plotInterval', 1,...            % interval between two call of "plotnsga".
... % Genetic algorithm operators
   'crossover', {{'intermediate',2}},...       % crossover operator (Ratio=1.2)
   'mutation', {{'gaussian',0.8,0.2}},...       % mutation operator (scale=0.1, shrink=0.5)
   'crossoverFraction', 0.9, ...             % crossover fraction of variables of an individual
   'mutationFraction', 0.5, ...              % mutation fraction of variables of an individual
... % Algorithm parameters
   'useParallel', 'yes',...                  % compute objective function of a population in parallel. {'yes','no'}
   'poolsize',4,...                         % number of workers use by parallel computation, 0 = auto select.
... % R-NSGA-II parameters
   ...%'refPoints', [],...                   % Reference point(s) used to specify preference. Each row is a reference point.
'refWeight', [0.5 0.5],...                   % weight factor used in the calculation of Euclidean distance
'refUseNormDistance', 'no',...               % use normalized Euclidean distance by maximum and minumum objectives possiable. {'front','ever','no'}
   'refEpsilon', 0.1 ...                     % parameter used in epsilon-based selection strategy
);
%***********************************************************************
```

**eedistrict_function.m (cost function)**

```
% this function is the cost function which the nsga2 optimisation invokes. It is in an adapted
version of the file presented earlier – eedistrict.m

function [objectives,co] = eedistrict_function(h_vec)

[heatProductionSchedule,padded] = vec2mat(h_vec,48);

%*************************************************************************

%{ the .m files below initialises all the constants and the variables needed for the analytical
model to run %}

eedistrict_variables;

eedistrict_constants;

%*************************************************************************

% reading 24-hour demand data for each building using "xlsread" function.

heatDemand = xlsread('heat_demand.xls');

electricityDemand = xlsread('electricity_demand.xls');

%*************************************************************************

% "total_demand" function is called to calculate the daily total heat energy and electricity
energy demand %}

[totalHeatDemand,totalElectricityDemand]=total_demand(heatDemand,electricityDemand);

%*************************************************************************

%Correction of schedules is shown below using function 'correction_schedule'

[heatProductionSchedule]=correction_schedule(GenerationUnitLowerBounds_Kwh,Generatio
nUnitUpperBounds_Kwh,heatProductionSchedule,nbofTimeslots,nboftypes)

%*************************************************************************

%..........rest of the code is same as  shown in eedistrict.m earlier....................................

end

%*************************************************************************
```

**correction_schedule.m**

```
% this function makes sure that energy production sources which are not needed to meet the
demand are turned off by changing the production schedule.

function
[heatProductionSchedule]=correction_schedule(GenerationUnitLowerBounds_Kwh,Generatio
nUnitUpperBounds_Kwh,heatProductionSchedule,nbofTimeslots,nboftypes,newDemand)

productionType= [1 0 0 0;2 3 0 0; 4 5 6 7];

m=size(productionType);

productionType_row=m(1);

productionType_column=m(2);
```

```
j=1;
type_sum =zeros(nboftypes,nbofTimeslots);
disp('newDemand')
 for g=1:nbofTimeslots
   for i=1:productionType_row
      for k=1:productionType_column
         if(productionType(i,k)~=0)
            type_sum(i,g)=type_sum(i,g)+heatProductionSchedule(productionType(i,k),g);
         end
       %  k=k+1;
      end
      %i=i+1;
   end
  %g=g+1;
 end
for m=1:nbofTimeslots
   n=1;
   while (n<=productionType_row)
      if (newDemand(m)>0)
         if(newDemand(m)> type_sum(n,m))
            newDemand(m)= newDemand(m)- type_sum(n,m);
         elseif(newDemand(m)<= type_sum(n,m))
            type_sum(n,m)= newDemand(m);
            newDemand(m)=0;
         else
            newDemand(m)=0;
         end
      else
         type_sum(n,m)=0;
      end
      n=n+1;
   end
```

```
end
for l=1:nbofTimeslots
   for i=1:productionType_row
      j=1;
      flag=1;
      availability=1;
      while(flag==1)
         if((productionType(i,j)~=0))
            while ( type_sum(i,l) >0&&availability==1)
               if(type_sum(i,l)>= GenerationUnitUpperBounds_Kwh(productionType(i,j)))

heatProductionSchedule(productionType(i,j),l)=GenerationUnitUpperBounds_Kwh(production
Type(i,j));
                  type_sum(i,l)=type_sum(i,l)-
GenerationUnitUpperBounds_Kwh(productionType(i,j));


               else if(type_sum(i,l)<=GenerationUnitLowerBounds_Kwh(productionType(i,j)))

heatProductionSchedule(productionType(i,j),l)=GenerationUnitLowerBounds_Kwh(production
Type(i,j));
                     type_sum(i,l)=0;
                  else
                     heatProductionSchedule(productionType(i,j),l)=type_sum(i,l);
                     type_sum(i,l)=0;
                  end
               end
               j=j+1;
               if(j<=productionType_column)


                  if((productionType(i,j)==0))
                     availability=0;
                  else
                     availability=1;
                  end
```

```
            end
          end
          if (j<=productionType_column)
            if((productionType(i,j)~=0))
              heatProductionSchedule(productionType(i,j),l)=0;
            end
          end
        end
        j=j+1;
        if(j>productionType_column)
          flag=0;
        end
      end
    end
  end
disp('new heat production');
disp (heatProductionSchedule);
%*************************************************************
```

*The following code presented in this appendix is detailed code. A simplified version of this code is also produced to combine the same type of energy sources into one which is not presented in this thesis.*