Project Acronym:	**BIMERR**
Project Full Title:	**BIM-based holistic tools for Energy-driven Renovation of existing Residences**
Grant Agreement:	**820621**
Project Duration:	**45 months**

DELIVERABLE D7.6
Building Energy Modelling Module 2

Deliverable Status:	**Final**
File Name:	**BIMERR_D7.6-v1.0**
Due Date:	**30/06/2020 (M30)**
Submission Date:	**30/06/2020 (M30)**
Task Leader:	**HYPERTECH (T7.3)**

Dissemination level	
Public	x
Confidential, only for members of the Consortium (including the Commission Services)	

The BIMERR project consortium is composed of:		
FIT	Fraunhofer Gesellschaft Zur Foerderung Der Angewandten Forschung E.V.	Germany
CERTH	Ethniko Kentro Erevnas Kai Technologikis Anaptyxis	Greece
UPM	Universidad Politecnica De Madrid	Spain
UBITECH	Ubitech Limited	Cyprus
SUITE5	Suite5 Data Intelligence Solutions Limited	Cyprus
HYPERTECH	Hypertech (Chaipertek) Anonymos Viomichaniki Emporiki Etaireia Pliroforikis Kai Neon Technologion	Greece
MERIT	Merit Consulting House Sprl	Belgium
XYLEM	Xylem Science And Technology Management Gmbh	Austria
CONKAT	Anonymos Etaireia Kataskevon Technikon Ergon, Emporikon Viomichanikonkai Nautiliakon Epicheiriseon Kon'kat	Greece
BOC	Boc Asset Management Gmbh	Austria
BX	Budimex Sa	Poland
UOP	University Of Peloponnese	Greece
UEDIN	University Of Edinburgh	United Kingdom
UCL	University College London	United Kingdom
NT	Novitech As	Slovakia
FER	Ferrovial Agroman S.A	Spain

Disclaimer

AUTHORS LIST

Leading Author (Editor)			
Surname	First Name	Beneficiary	Contact email
Giannakis	Giorgos	HYPERTECH	g.giannakis@hypertech.gr
Co-authors (in alphabetic order)			

#	Surname	First Name	Beneficiary	Contact email
1	Fenz	Stefan	XYLEM	fenz@xylem-technologies.com
2	Galanos	Zeppos	CONKAT	zgalanos@konkat.gr
3	Heurix	Johannes	XYLEM	heurix@xylem.tech
4	Kalamaris	Thanos	HYPERTECH	t.kalamaris@hypertech.gr
5	Katsos	Marios	HYPERTECH	m.katsos@hypertech.gr
6	Petridis	Kosmas	HYPERTECH	k.petridis@hypertech.gr
7	Sideris	Dimitris	HYPERTECH	d.sideris@hypertech.gr
8	Tsiakoumi	Matina	HYPERTECH	m.tsiakoumi@hypertech.gr
9	Papapolyzos	Dinos	HYPERTECH	d.papolyzos@hypertech.gr
10	Papapolyzos	Thomas	HYPERTECH	thomas@hypertech.gr

REVIEWERS LIST

List of Reviewers (in alphabetic order)			
#	Surname	First Name	Beneficiary	Contact email
1	Poveda-Villalón	María	UPM	mpoveda@fi.upm.es
2	Varga	Ján	NT	varga@novitechgroup.sk
3	Straka	Martin	NT	straka@novitechgroup.sk

REVISION CONTROL

Version	Author	Date	Status
0.10	HYPERTECH	29.04.2021	Table of Contents
0.20	HYPERTECH	10.05.2021	Section 1
0.30	HYPERTECH	17.05.2021	Section 3 Draft
0.40	HYPERTECH	24.05.2021	Sections 3.1, 3.2, 3.3
0.50	HYEPRTECH, XYLEM, CONKAT	27.05.2021	Section 2 – updates from D7.5
0.60	HYPERTECH, XYLEM	15.06.2021	Sections 3.4 – 3.9, 4

0.70	HYPERTECH	17.06.2021	Quality Check
0.80	NT, UPM	25.06.2021	Final Draft reviewed
1.0	HYPERTECH, FIT	30.06.2021	Submission to the EC

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

Acronym	Meaning
AC	Air Conditioning
AMY	Annual Meteorological Year
API	Application Programming Interface
BCVTB	Building Controls Virtual Test Bed
BEP	Building Energy Performance
BIF	BIMERR Interoperability Framework
BIMERR	BIM-based holistic tools for Energy-driven Renovation of existing Residences
BIM	Building Information Modelling
BSD	Building Surface Detailed
CBIP	Common Boundary Intersection Projection
CFD	Computational Fluid Dynamics
CSV	Comma Separated Values
EMS	Energy Management System
EPBD	Energy Performance of Building Directive
EPW	EnergyPlus Weather file
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
FSD	Fenestration Surface Detailed
GID	Global Identifier
HTTP	Hypertext Transfer Protocol
HVAC	Heating Ventilation and Air Conditioning
IDD	Input Data Dictionary
IDF	Input Data File
IFC	Industry Foundation Classes
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
obXML	Occupant behaviour eXtensible Markup Language
PRUBS	Life Cycle Costing
RenoDSS	Renovation Decision Support System
RDD	Report Data Dictionary

TMY	Typical Meteorological Year
VRF	Variable Refrigerant Flow

EXECUTIVE SUMMARY

This deliverable addresses the topic of how the Energy Key Performance Indicators (KPIs), introduced in D3.3 and refined in the context of T9.1 – *Design of Demonstration Activities, Ex-ante Renovation Project Analysis & Evaluation Baseline Establishment*, are to be computed and communicated to the Renovation Decision Support System (RenoDSS). The calculation of these KPIs is a key part of the process towards the identification of the best renovation scenarios according to stakeholders' preferences.

It constitutes the final version of the Building Energy Performance (BEP) module documentation, where results of a survey on existing BEP simulation methodologies, the BEP simulation engine's selection, delivered data pre-processing, mapping and post-processing functionalities, the technology stack that the BEP module builds on, the API that is exposed to RenoDSS, and application examples, are reported.

Unified building and occupant behaviour data models, based on the Industry Foundation Classes (IFC) and the occupant behaviour eXtensible Markup Language (obXML) standards, respectively, provide data for extracting relevant information to setup the BEP[1] simulations. Interfaces, aimed at generating respective inputs and invoking relevant external simulation engines (e.g., EnergyPlus), and ways of post-processing the results, are integrated in the BEP module so that the predefined Energy KPIs can be computed.

Since this work reflects the activities that have been undertaken towards delivering the final version of the BEP module, the functionalities that have been reported in the first version of the module's documentation are supplemented by new functionalities, and refinements of existing ones, that are delivered towards addressing assumptions and restrictions that had previously been identified. Feedback acquired during the BIMERR integration, pre-validation and validation activities, has played a pivotal role in the elicitation of requirements for extensions and new functionalities.

---

[1] Although the shortened form of the Building Energy Modelling module should be BEM, in literature, the BEM abbreviation is frequently referred to Building Energy Management systems. Hence, from this point forward, the Building Energy Performance (BEP) term is adopted.

1. INTRODUCTION

1.1 SCOPE AND OBJECTIVES OF THE DELIVERABLE

This deliverable aims at reporting on the work that has been conducted in the context of T7.3 – *Building Energy Performance Modelling Module* [1]. In alignment with the BIMERR architecture [2], the BEP module is considered as a sub-component of the BIMERR's Renovation Decision Support System, RenoDSS. In principle, RenoDSS provides to the renovation designer an accurate estimation of the Energy, Cost, Life Cycle Assessment trade-offs of various alternative renovation scenarios, based on the available renovation options in terms of materials and components.

The BEP module is responsible for the performance evaluation of the renovation measures in terms of specific Energy KPIs calculation that have been defined and documented in D3.3 – *BIMERR evaluation methodology* [3], and revised in the context of T9.1 activities.

The BEP module's core functionalities are to map the IFC [4] data to proper Input Data Format (IDF) files that are used to execute BEP simulations and combine them with obXML data for occupancy [5] to enable the RenoDSS users to explore various what-if scenarios, quickly run simulations to estimate the energy performance and fine-tune the interventions in order to explore the benefits. The need for the aforementioned functionalities stems from the fact that within BIMERR, Industry Foundation Classes (IFC) files are used to streamline and expedite the collection of the building static information (building elements, materials and their thermal properties, HVAC components, to name but a few), while obXML files capture the dynamic data (schedules and thermal comfort preferences), generated by the Profiling Residents Usage of Building Systems (PRUBS) tool [6]. Since a wide range of BEP simulation methodologies exists, a survey on existing BEP calculation methodologies that exploit both IFC and obXML data to automate the BEP input data files' generation, and the selection of the BEP simulation engine that best fits the BIMERR case-studies, has been the preliminary objective of T7.3, whose results have also been documented in D7.5 [7]; for the sake of completeness, they are also reported in this document.

The main aim of this deliverable is to provide a comprehensive overview and documentation of the BEP module, and as a "Demonstrator" type deliverable, to report on the actual service that has been developed and delivered.

1.2 RELATION TO OTHER TASKS/DELIVERABLES

Table 1 depicts the relations of this document to other deliverables within the BIMERR project, that should be considered along with this document for further understanding of its contents.

Table 1 Relation to other BIMERR project's deliverables

Del. Number	Deliverable Title	Relations and Contribution
D3.1 [8]	Stakeholder requirements for the BIMERR system	Analysis of the end-user requirements in order to create the necessary inputs for defining the different components of the BIMERR, along with a thorough description of the business scenarios, use cases and system requirements tailored to the project's goals and therefore setting the skeleton for the BIMERR framework.
D3.3 [3]	BIMERR evaluation methodology	The evaluation methodology of the project based on well-established international methodologies and protocols, reviewing the state of the art and relevant BIM projects, as well as relevant renovation works and expertise of the BIMERR constructor partners; Documentation of the energy KPIs.
D3.6 [2]	BIMERR system architecture final version	The final version of the BIMERR architecture is delivered. The structural view, describing the core components of the system in the form of software modules, and the dynamic view, which presents the already defined use cases with the corresponding sequence diagrams are described highlighting the data exchange and interoperability requirements of each tool.
D4.3 [9]	BIMERR Ontology and Data Model 2	The final version of the BIMERR ontologies and data models' structure is developed to address the various semantic interoperability challenges for BIM-related data in an efficient manner.
D5.2 [10]	Prototype of enhanced BIM platform 2	The final version of the enhanced BIM platform is documented, providing information about the IFC content's schema compatibility, correctness, and completeness.
D5.4 [11]	Innovative Scan-to-BIM tools for Automated BIM generation 2	The final version of the Scan-to-BIM tool and its sub-components is delivered, documenting the pathway to enrich the IFC file with the 2nd level space boundaries topology, material properties and energy related properties of the HVAC components.
D5.8 [6]	Building resident energy related behaviour profiling framework 2	The final version of the PRUBS component that provides information about how the dynamic data, required as input to the BEP module, are being population and provided following the obXML schema.
D7.5 [7]	Building Energy Modelling Module 1	The first version of BEP module's documentation is provided that has constituted the bedrock of this document.

D8.3 [12]	Integrated BIMERR ICT system 1	The first version of the end-to-end ICT system integration, testing and refinement, where information about the BIMERR components version control, delivery and deployment.

Functionalities that are introduced in this document are deployed to also address the aspects relevant to the scope of T7.5 – *Decision Support System Engine and UI & Module Integration* [1], where continuous integration of all WP7 components and testing are performed to ensure that functional and non-functional requirements of the Renovation Decision Support System are met. Note here that the output of the BEP module affects the calculation of other components of RenoDSS. Thus, the content of this document is relevant to all deliverables of WP7.

1.3 UPDATES TO THE FIRST VERSION OF THE BUILDING ENERGY PERFORMANCE MODULE

The focus of this deliverable has been to report on refinements and additions to the functionalities of the module that have already been documented in D7.5, based on verification and validation experiments performed in relevant environments (pre-validation pilot sites). Hence, this document complements D7.5, providing information about actions that have been taken to address the assumptions and restrictions that have been reported in D7.5. These actions are listed below:

- Material thermal properties – IfcPropertySets have been developed so that relevant information can be captured in IFC; the relevant properties are tailored to the properties sets required for the Material, Material:NoMass and Window MaterialSimpleGlazingSytem objects definition, thus allowing an IFC to IDF one-to-one mapping to be performed (see Section 3.1.3).
- HVAC demand and supply side components – Additional IfcPropertySets have been developed to capture the HVAC energy related data for types of systems that correspond to the actual HVAC system of the pre-validation and validation sites. Revit[2] families have also been created so that exported IFC files set the basis on how such components should be handled by Scan-to-BIM. Hence, the respective HVAC input data requirements are met. Rules for proper HVAC components' IFC to

---

[2] Introduction to Revit families

IDF mapping have been implemented in the Input Data File Generator (see Section 3.1.4).

- Renewables – Taking into account the characteristics of the pre-validation and validation sites, in the second release of the BEP module, the simulation of Photovoltaics (PVs) and Solar Collectors is supported. Prerequisite for their simulation has been the identification of the IFC classes and the development of the properties sets that are used to capture relevant information. Such information has been considered to create Revit families for their modelling. Rules for a seamless renewable systems' IFC to IDF mapping have been implemented in the Input Data File Generator, presented in Section 3.1.5.

- Data exchange with the RenoDSS data management module – In the first version of the BEP module, the communication with the RenoDSS data management module was synchronous. However, since numerous candidate renovation scenarios evaluation requests are sent in parallel, the use of asynchronous communication tools to distribute computations and to gather results is supported in the second release of the module.

- EnergyPlus[3] and ObXML Co-Simulation – In view of the fact that the FMUs/FMI based co-simulation setup, that has been delivered as part of the first release of the BEP module, significantly increases the simulation runtime, the implementation and the performance evaluation of an alternative approach that utilises the EnergyPlus Python Data Exchange API, supported by the latest versions of EnergyPlus, is delivered as an option for the obXML and IDF data-exchange.

1.4   STRUCTURE OF THE DOCUMENT

To address the aspects relevant to the scope of T7.3, Section 1 introduces the work performed and the scope of this deliverable, along with its relevance to other BIMERR tasks and the deliverable's structure. Section 1.3 highlights the changes that have been performed, compared to the first release of the BEP module.

---

[3] https://energyplus.net/

As a short introduction to the BIMERR BEP module requirements, (1) a survey on existing methodologies and widely used simulation engines towards selecting the BEP simulation engine that best fits the BIMERR objectives, (2) a summary of the BIMERR Energy KPIs, and the inputs/outputs formats of the selected engine, are reported in Section 2.

Section 3 summarises the technical work that has been conducted up to M30 within T7.3. Having concluded to the BEP module architecture, a transformation process that converts the data obtained from the IFC to the BEP simulation engine's input data file is required; rules embedded in the "Input Data File Generator", responsible for that transformation process, are presented in Section 3.1. Establishing a link between the BEP input data file and the occupant behaviour data model (obXML), so that dynamic schedules can be incorporated in the simulation, is of paramount importance towards increasing the energy performance estimation accuracy. This functionality, which we refer to as co-simulation, is introduced in Section 3.2. Beyond a wide range of the BEP simulation output data and files, particular results are relevant and need to be further processed to estimate the Energy KPIs. Such data post-processing functionalities are reported in Section 3.3. Application examples and results are presented in Section 3.8. The remaining parts of Section 3 present the technology stack, the API documentation, assumptions, restrictions, and licensing that have been considered in the second release of the BEP module. Finally, in Section 4, conclusions are provided.

2. ENERGY KPIS AND BEP SIMULATION ENGINE SELECTION

According to the BIMERR architecture [2], the BEP module is invoked by the RenoDSS tool to evaluate the as-is thermal behaviour of a building (baseline) and estimate the energy performance of certain renovation scenarios, that are candidates to be applied. To deliver its scope, the BEP module retrieves information from the RenoDSS management module (i.e. IFC, obXML and EPW files), transforms them to proper simulation input data files and launches respective simulation tools (or engines) to calculate predefined KPIs.

Towards achieving the data retrieval, transformation, and appropriate simulation tools execution, the following points for discussion have arisen: Which are the KPIs that will be calculated for the ranking of different renovation scenarios? Since simulation of numerous different renovation scenarios will be requested, which is the BEP simulation engine that strikes a balance between accuracy and computational complexity? Do the simulation outputs require any post-processing before being communicated back to RenoDSS? In the following paragraphs, after a prompt to the energy KPIs that have been analysed in D3.3 [3], these questions are answered.

2.1 ENERGY KEY PERFORMANCE INDICATORS

D3.3 has focused on the definition of the BIMERR Key Performance Indicators that are to be used in BIMERR to evaluate the performance of candidate renovation measures and is strongly linked to this deliverable. For the sake of completeness, a summary of the Energy KPIs that the BEP module must be capable to calculate is listed in Table 2.

Table 2 List of Energy Key Performance Indicators according to the BIMERR evaluation methodology [3]

KPI	Name	Definition/Description	Units
EN1	Total primary energy consumption	Primary energy consumption refers to the direct use at the source, or supply to users without transformation, of crude energy, that is, energy that has not been subjected to any conversion or transformation process	kWh/m^2/year
EN2	PENRT Primary energy non-renewable total	The primary energy content of all non-renewable resources (crude oil, coal, etc.).	kWh/m^2/year

EN3	Electric energy consumption	The building's total electric energy consumed for the operation of HVAC systems, lighting, and appliances	kWh/m^2/year
EN4	Natural gas energy consumption	The building's total natural gas energy consumed for the operation of HVAC systems and appliances	kWh/m^2/year
EN5	District heating energy consumption	The building's total district heating energy consumed for the operation of HVAC systems and domestic hot water equipment	kWh/m^2/year
EN6	Other fuel types consumption	The building's total energy consumed, generated by other resources, e.g. diesel, biomass energy consumption	kWh/m^2/year
EN7	Peak heating load and heating load profile	The heating power profile for a specific period in Watts and its maximum/peak value	Watts
EN8	Peak cooling load and cooling load profile	The cooling power profile for a specific period in Watts and its maximum/peak value	Watts
EN9	Heating and cooling energy demand	The sum of heating and cooling loads required to maintain the thermal comfort conditions to all the conditioned building's spaces	kWh/m^2/year
EN10	Peak electricity load and electricity load profile	The electricity power profile for a specific period in Watts and its maximum/peak value	Watts
EN11	PV electric energy generation	The electric energy generated by photovoltaic panels installed on the building's envelope	kWh/m^2/year
EN12	Solar thermal energy generation	The thermal energy generated by solar thermal panels installed in the building	kWh/m^2/year

2.2 BEP SIMULATION METHODOLOGIES AND ENGINES

The evaluation of the Energy KPIs is a complex, multi-parametric problem, being solved by different methods that fall under two main categories: quasi-steady state and dynamic methods.

According to [13], the monthly-based calculation methodology described in ISO 13790:2008 [14] standard is a prime example of a quasi-steady state method, adapted by many EU member states to form at a national level an accepted calculation methodology for computing energy performance, in the context of activities for the implementation of the Energy Performance of Buildings Directive (EPBD) [15]. As part of that standard, the accuracy of the calculation methodology that it adopts is analysed, indicating certain scenarios where deviations from reality are acceptable in an annual

basis, but on a monthly scale these deviations are significant. The sensitivity to input data is also reported: uncertainties in the estimation of thermal properties or other input parameters can contaminate the results, and the propagation of these errors can yield sizable deviations in the end results. For this reason, in many cases, the calculation methodology is used to establish an ordering relation, that allows for meaningful comparisons of different retrofitting scenarios (and thus establishing the rating system used in many countries), but with lesser expectations with regards to prediction of real performance.

Within BIMERR, the BEP module aims to enhance the Decision Support System by providing an accurate estimation of post-renovation building energy consumption. To deliver its scope, the BEP module takes into account the followings: (a) structural and geometrical properties of the building; (b) materials; (c) heating/cooling system and other relevant energy loads; and (d) most importantly information about the actual usage patterns of these loads based on the resident behavioural patterns and comfort preferences within the home (obXML data, generated by the PRUBS component), whose nature is strictly dynamic. In other words, dynamic effects matter, and the temporal resolution of a month that the steady-state methods imply is not sufficient.

Dynamic methods follow a more granular approach, using smaller time steps which are comparable to the time scale of time-varying physical effects that are being modelled, e.g. the conductive and convective heat transfer from building thermal masses or the dynamic effects of the HVAC system's operation. In their highest granularity, dynamic methods touch upon the Computational Fluid Dynamics (CFD) topic. CFD for whole building energy performance simulation can be helpful as a quasi-qualitative tool for understanding fundamental flow structures. However, their computational complexity (in terms of simulation runtime) and the requested domain expertise are almost prohibitive for use in energy renovation studies, where several candidate renovation scenarios need to be simulated within a reasonable timescale and on the basis of realistic assumptions. Accordingly, the use of CFD calculation methodologies is mostly restricted in the design phases of a building life cycle.

Other dynamic methods introduce the thermal zones concept, where the building is divided into spaces (thermal zones), each with a uniform temperature, humidity etc. The

3D thermal zones-type methods have gained ground in the BEP simulation domain, being used in many envisaged and practical use scenarios, as it keeps accuracy and computational complexity in balance. Beyond a wide range of existing thermal zones-type BEP simulation engines, DeST [16], EnergyPlus [17], eQUEST, ESP-r, IES-VE [18], TRNSYS [19], and Modelica [20], are the most frequently used, but all are suffering from two major drawbacks: (1) their input data files' preparation is a very time-consuming process due to the difficulty to collect relevant information, often requiring more time than it is available due to project's deadlines, and (2) it is a non-standardized process that produces BEP simulation models whose results can significantly vary from one modeler to another according to their experience.

Within BIMERR, IFC and obXML files are used to streamline and expedite the collection of such information, while our BEP approach intend to introduce a methodology to automatically translate IFC to BEP simulation input data, making the BEP simulation modelling process much more expedient and less vulnerable to modelling errors. The topic of automated data translation between BIM and thermal simulation input data has received considerable attention as of late [21, 22, 23, 24, 25], where studies have shown that EnergyPlus and Modelica consist the most frequent used engines. Furthermore, within the EBC-Annex 66 [26] project, an overview of the thermal zones-type engines capabilities to implement stochastics occupant behaviour models, provided through the obXML entities, has been performed, indicating the EnergyPlus flexibility on capturing such data in three different ways, briefly presented in Section 2.3.

Due to our preference for the implementation of an open-source BEP simulation engine, EnergyPlus has been selected; robust Modelica libraries are commercial, while a free open-source library of Modelica models for BEP simulation is under development, as part of IBPSA Project-1 workplan[4].

---

[4] https://ibpsa.github.io/project1/index.html

2.3   BEP Simulation – EnergyPlus

EnergyPlus is a software released by the U.S. Department of Energy. EnergyPlus follows the thermal zoning, spatial-discretization, paradigm, where the building is divided into spaces (thermal zones), representing nodes with averaged values of thermal parameters. The energy conservation differential equation and the mass conservation differential equation hold for each zone and are solved to evaluate the evolution in time of the zonal thermal parameters.

The main input file is the Input Data File (IDF), an ASCII file which contains information about the building and the HVAC system to be simulated. The EnergyPlus input data are structured into classes. For each class, fields are defined, which describe the characteristics of the class objects. Objects are the instances of a class. All the available classes are listed into the Input Data Dictionary (IDD)[5]. The EnergyPlus Weather file (EPW)[6] is a Comma Separated Value (CSV) format file containing the hourly or sub-hourly weather data needed by the simulation program.

Concerning the obXML data insertion, there are three main groups of IDD classes, whose objects could be used to incorporate occupant behaviour data: (1) Energy Management System (EMS) classes that allow custom functions to overwrite the occupants' presence and actions data (schedules) at each simulation timestep; (2) Building Controls Virtual Test Bed (BCVTB) API classes that enable the co-simulation of EnergyPlus with an external tool, where the occupant behaviour modelling functions are simulated; and (3) Function Mockup Units (FMUs) classes that enable the aforementioned co-simulation in a more efficient manner. As of EnergyPlus version 9.3, a new functionality of EnergyPlus to overcome the EMS scripting complexity by writing scripts in Python programming language that interact with the certain elements of the IDF has become available. For instance, the content of the obXML can be automatically translated to Python scripts that are used for the obXML and IDF data exchange.

---

[5] bigladdersoftware - EnergyPlus 9.3 - Input Data Dictionary (IDD)

[6] bigladdersoftware - EnergyPlus 9.3 - EnergyPlus Weather File (EPW) Data Dictionary

Beyond a wide variety of EnergyPlus output variables, particular variables can be reported depending on the actual simulation problem described in the IDF. The Report Data Dictionary (RDD)[7] is a text file listing those variables available for reporting during the simulation of a certain IDF. Selecting an output variable from that list, an object of the Output:Variable class is defined and imported in the initial IDF.

After an initial IDF – enriched with the selected Output:Variable objects – simulation run, the resulted data-sets of the selected variables are printed in a comma separated text by a semi column, where each column corresponds to a unique variable time-series with the reporting frequency defined by the modeller, commonly equal to the simulation timestep.

---

[7] bigladdersoftware - EnergyPlus 9.3 - Report Data Dictionary (RDD)

3. BIMERR BUILDING ENERGY PERFORMANCE MODULE

Considering the input data requirements and the results' reporting capabilities of EnergyPlus, the BEP module prototype has been designed and developed, consisting of three main sub-components (see Figure 1): IDF Generator; BEP simulation; and BEP Manager.
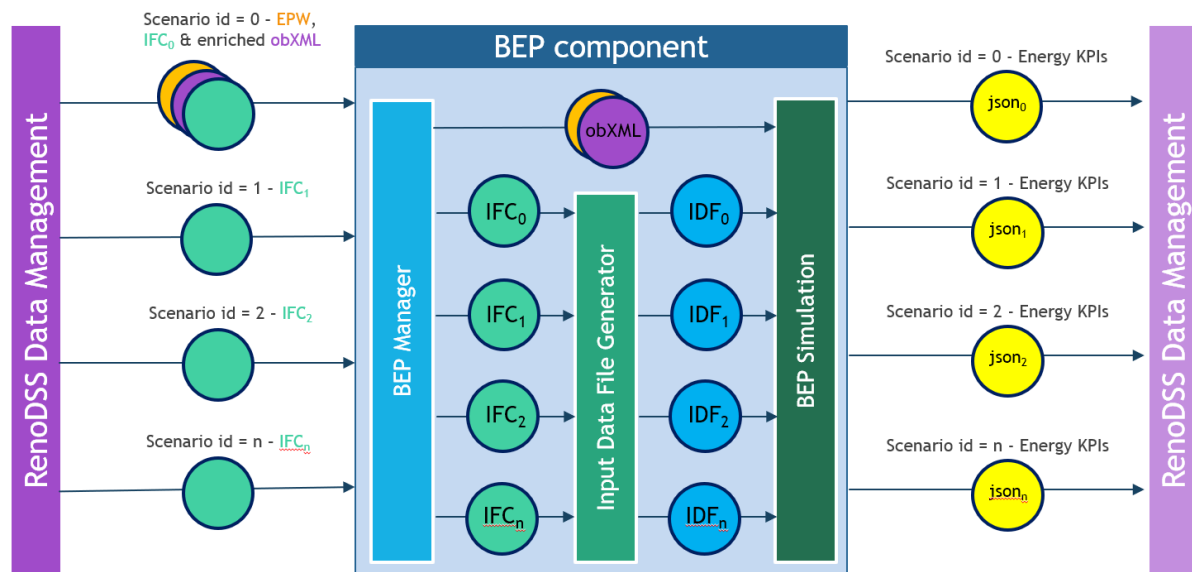
Figure 1 – Architecture of the BEP module.

Analysing the energy KPIs calculation process, whenever the BEP Manager receives a scenario evaluation request from the RenoDSS data Management module (in terms of energy KPIs calculation), the IDF Generator module is invoked. We assume that each scenario has a unique ID, while the scenario with ID value equal to 0 corresponds to the baseline. Based on this assumption, when the baseline evaluation is requested, a process of 6 steps is initiated.

As Figure 2 illustrates, the IDF Generator sends a request for the relevant IFC, obXML and EPW files to the RenoDSS Data Management module. Retrieving the aforementioned from the RenoDSS Data Management module, the IDF generator is processing the content of the IFC file to generate the Input Data File of EnergyPlus simulation engine, while the co-simulation with the obXML, to take into account actual schedules based on data-driven occupant behaviour models, is being established. Then, the input data requirements for

an EnergyPlus simulation are met, and the IDF Generator submits a simulation request to the BEP simulation module. When the simulation is completed, outputs/report files are populated and sent back to the IDF Generator to be forwarded to the BEP Manager.
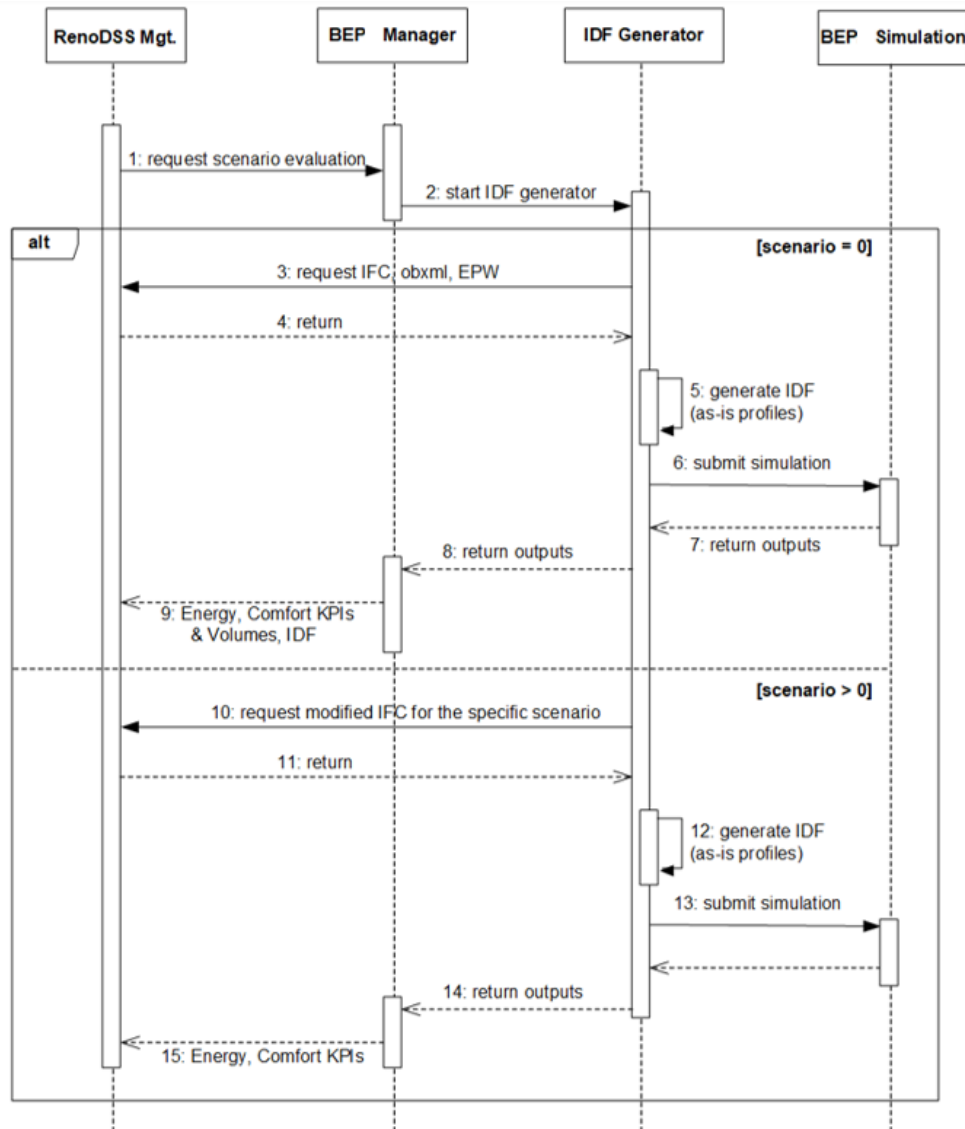
Figure 2 – Sequence diagram of BEP module execution for the Energy KPI calculation - baseline (scenario = 0) and renovation scenarios (scenario > 0) [2].

Finally, the outputs/report files are processed within the BEP Manager to populate a JSON file with the KPIs that are sent to the RenoDSS Data Management module as the result of the baseline evaluation request.

An almost identical process is followed to handle requests of candidate renovation scenarios evaluation. The main difference occurs in the files requested and processed by the BEP module: since the renovation measures affect and modify objects of the IFC file only, there is no need to resend the obXML and EPW files.

In summary, the final version of the BEP module provides the following functionalities, as described in Sections 3.1, 3.2, and 3.3:

- It retrieves IFC, EPW and obXML data from RenoDSS for a specific scenario and applies a transformation process to populate the input data file of EnergyPlus.
- It automatically generates the configuration files requested for the co-simulation setup of EnergyPlus and obXML, based on either the Functional Mock-up Interfaces standard (FMIs) or utilising the EnergyPlus Python Data Transfer API.
- Using the EnergyPlus and obXML Functional Mock-up Units (FMUs) as input, the BEP module automatically invokes the co-simulation execution. In the alternative option, neither EnergyPlus IDF nor obXML need to be exported as FMUs, decreasing the co-simulation setup time significantly.
- When the simulation run has been finished, the BEP module further processes the simulation output files and generates a JSON file as an output message (to be listened by the RenoDSS) that contains the energy KPIs values for the specific renovation scenario.

3.1 ENERGYPLUS INPUT DATA FILE GENERATION

The Input Data File Generator constitutes the core component of the BEP module. Utilising the IFC and obXML data, it generates the BEP simulation Input Data File (IDF) of EnergyPlus.

The input data of EnergyPlus can be grouped in three main categories: (1) static data that include the building geometry, construction materials, glazing information, systems used in the building, building's spatial discretization to thermal zones, to name but a few; (2) dynamic data that consist of all time-variant data, such as user-actions, occupancy schedules at each thermal zone, use of equipment, etc; and (3) other simulation

parameters that refer to features requiring domain expertise for input specification and output assessment, and cannot be captured by IFC and/or obXML.

In the following subsections, the BEP input data, required for an EnergyPlus simulation, are briefly described, while the rules that the Input Data File Generator encapsulates to automatically generate the IDF data and the co-simulation set-up, utilising the IFC and obXML data, are presented.

3.1.1 Building Discretization to Thermal Zones

As introduced in Section 2.3, a principal modelling assumption of EnergyPlus is the spatial discretisation of the building volume to thermal zones. In EnergyPlus, thermal zones are defined by instances of the Zone[8] class. Thermal zones are not geometrically represented, but a geometric representation is being provided for each surface that bounds them (see Section 3.1.2). In other words, a thermal zone can be seen as a group of surfaces that bound it. Such a group is being displayed in the right side of Figure 3.
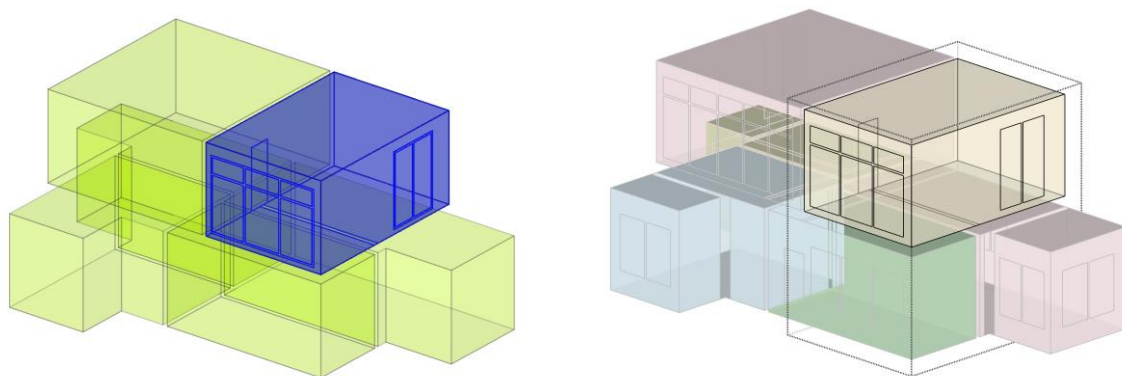
Figure 3 – A thermal zone box consisting of a group of surfaces in IFC (left) and IDF (right), rendered by Solibri Viewer and OpenStudio plugin for Sketchup, respectively.

Although IFC supports information capturing about the thermal zone's definition through instances of the IfcSpatialZone class, it cannot relate them with surfaces that express the boundary conditions to which the thermal zone is exposed.

[8] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - Zone

> **Rule 1**: To overcome such a limitation, each space is treated as a thermal zone. Hence, to define the thermal zones, the building's spaces, defined as IfcSpace[9] class objects, are queried and mapped to Zone IDD class objects. Note here that although the IfcSpatialZone class is not used for the thermal zones' definition, within BIMERR, this class has been selected to represent apartments of a building as groups of spaces.

3.1.2 Building Elements – Geometric Representation

To describe the building geometry, the type of boundary condition applied to each building element and the construction assigned to it, EnergyPlus IDD provides several classes to define surfaces, ranging from simple rectangular surfaces to detailed descriptions, with the latest being frequently used. For the detailed description, BuildingSurface:Detailed[10] (BSD) and FenestrationSurface:Detailed[11] (FSD) classes are recommended. BuildingSurface:Detailed class is used to describe opaque building surfaces, while FenestrationSurface:Detailed class is used to describe transparent building surfaces. Their geometry representation can be described by planar surfaces only (a set of vertices in a specific order).

In terms of the IFC schema, relevant information is captured by instances of the IfcRelSpaceBoundary2ndLevel (SB2L) class (see Figure 4), whose connection geometry is restricted to planar surfaces only, thus aligned with the geometry representation requirements of EnergyPlus.

Each SB2L is translated to a BSD or a FSD object as follows:

> **Rule 2**: BSD – IF the SB2L does not have a Parent Boundary, THEN it is translated to a BSD object, whose Name and Zone Name properties values are equal to the space boundary GID and the space boundary Relating Space GID, respectively. For the Surface Type

---

[9]https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/link/ifcspace.htm

[10]bigladdersoftware - EnergyPlus 9.3 - I/O Reference - BuildingSurface:Detailed

[11]bigladdersoftware - EnergyPlus 9.3 - I/O Reference - FunestrationSurface:Detailed

definition, the Related Building Element, the Corresponding Space Boundary, and the normal vector data of the SB2L are processed. Moreover, IF SB2L is external, THEN for the generated BSD the following hold: Outside Boundary Condition, Sun Exposure, and Wind Exposure properties values are set to Outdoors, Sun Exposed and Wind Exposed, respectively; else, the Outside Boundary Condition value is set to Surface, and the Outside Boundary Condition Object is the corresponding space boundary GID. Completing the transformation to a BSD class object, Number of Vertices and coordinates of each vertex are fulfilled according to the space boundary geometric representation.
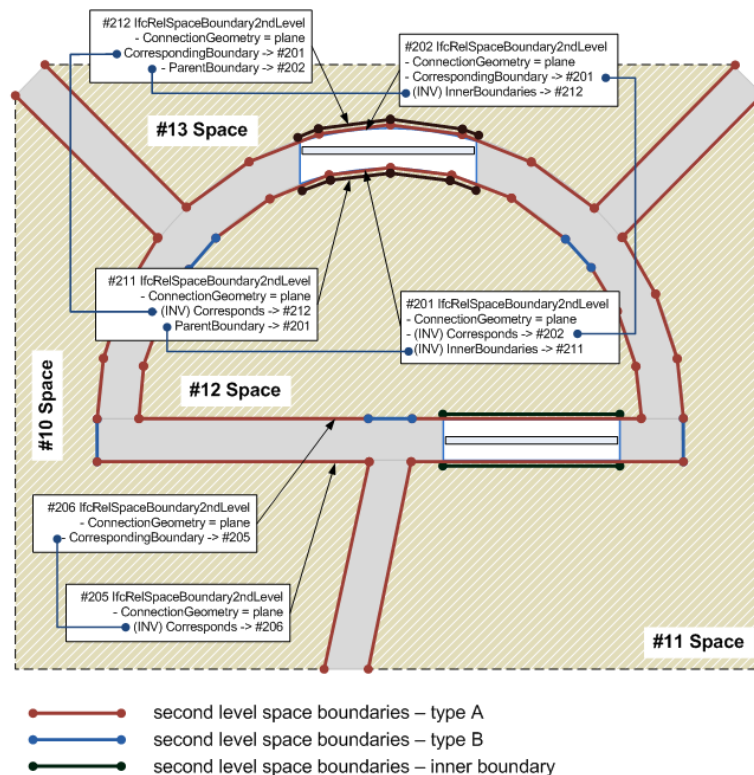
Figure 4 – An illustration of the 2nd-level space boundaries relationships[12].

Rule 3: FSD – IF the SB2L has a Parent Boundary, THEN it is translated to a FSD object, whose Building Surface Name is the parent boundary surface GID. If its Corresponding space boundary has a relating space, the Outside Boundary Condition Object is the corresponding boundary GID. IF the SB2L's construction consists of opaque Materials,

---

THEN the Surface Type of the FSD object is Door; else the Surface Type is Window. Completing the transformation to a FSD class object, Number of Vertices and coordinates of each vertex are fulfilled according to the space boundary geometric representation.

Given the 2D representation of the 2nd-level space boundaries, a set of rules are applied to transform the 2D to a 3D representation, utilising the different levels of relative placements that are provided by the IfcPlacement[13] objects, which define the location and orientation of an item.

The results of these mapping rules applied to two SB2L, one having roof and one having window as related building elements, are illustrated in Figure 5.
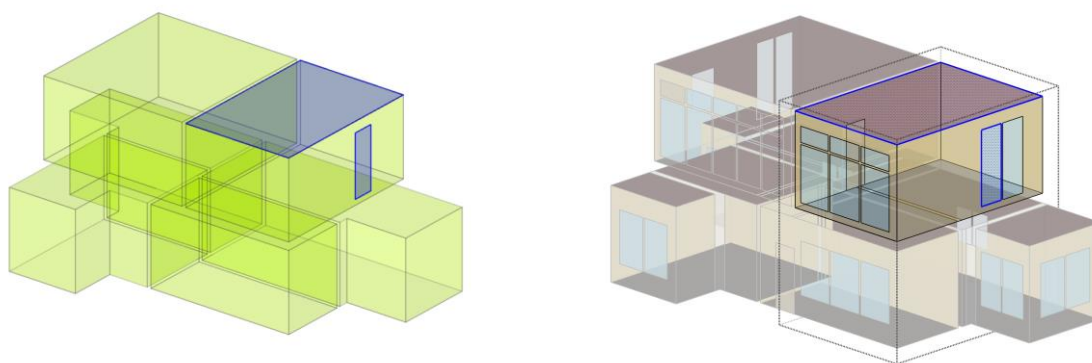
Figure 5 – From the 2nd-level space boundaries (left – Solibri Viewer) to the BuildingSurface:Detailed and FenestrationSurface:Detailed (right – OpenStudio plugin for Sketchup).

It is worth mentioning that although the 2nd-level space boundary data correctness is prerequisite, the IFC files often contain data that are missing or incomplete due to design errors or exporting software imperfections. Hence, a consistent way of capturing the 2nd-level space boundary topology is of paramount importance. In view of this, an algorithm that generates that topology is being developed within T5.2 [11]. In the same context, a 2nd-level space boundary generation tool, CBIP, has recently been developed [27], to calculate the 2nd-level space boundary surfaces based on the geometrical information of the architectural elements (walls, spaces, slabs, openings, etc.) contained in its input IFC file and enrich it by populating appropriate IFC data classes in its output. The CBIP tool

---

[13] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/link/ifcplacement.htm

has been implemented in the BIMERR BIM platform [10], providing the option to populate and enhancing the IFC data with the 2nd-level space boundary topology.

3.1.3   Building Elements – Materials Thermal Properties and Construction Data

The construction of each structural building element (e.g., wall, slab, column) is composed of material layers. To define its material layers' bedding in relevant order, EnergyPlus introduces the Construction class. In IFC, each Construction that refers to a structural building element is defined by an IfcMaterialLayerSet [14] object, composed by IfcMaterialLayers. Conversely, the construction of each opening element (door, window) is represented by a single layer, whose information is defined by type (objects of the IfcWindowType[15] and IfcDoorType[16] classes). To objectify the relationship between a material layer set and a structural building element, the IfcRelAssociatesMaterial class is used.

> **Rule 4, 5**: Two rules are applied to populate the objects of the Construction class of EnergyPlus: (rule 4) for each IfcWindowType or IfcDoorType object, a Construction object with a single layer is generated; and (rule 5) for each IfcMaterialLayerSet, a Construction object with layers equal to the MaterialLayers attribute's value of the IfcMaterialLayerSet object is generated.

In Figure 6, the results of these rules for different IfcMaterialLayerSets, IfcWindowType and IfcDoorType objects are displayed.

Delving deeper into the material and its thermal properties definition, EnergyPlus provides several classes to describe materials and their thermal characteristics, with the Material, Material:NoMass, and WindowMaterial:SimpleGlazingSystem, being widely

---

[14] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/ifcmateriallayerset.htm

[15] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/ifcwindowtype.htm

[16] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/ifcdoortype.htm

used. In terms of IFC, IfcMaterial, IfcMaterialLayer [17], IfcMaterialProperties [18] and IfcPropertyPset[19] classes are used to capture relevant information.
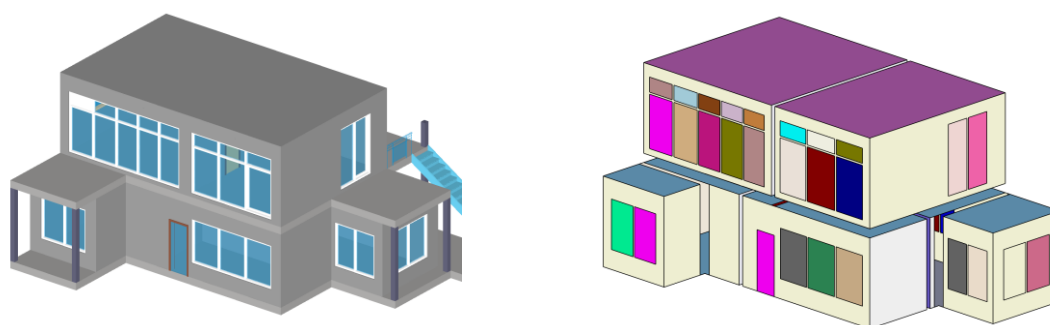
Figure 6 – From structural and opening elements (left – Solibri Viewer) to Construction objects of EnergyPlus (right – render objects by Construction using the OpenStudio plugin for Sketchup).

Within BIMERR, the RenoDSS provides functionalities for the IFC file's enrichment with thermal properties values of each material, utilising the material and components database as the main data source. For the baseline IFC, such data become available, exported either from the BIM authoring tool by modifying its IFC exporter plugin [10] or populated by Scan-to-BIM [11]. To that direction, property sets (IfcPropertySet), tailored to the scope of capturing material thermal properties, have been developed. The IfcPropertySet is a container that holds properties within a property tree. These properties are interpreted according to their name attribute. Each individual property has a name string. For these and other property sets that have been defined within BIMERR, the naming convention "Pset_Xxx" that applies to all predefined property sets that are provided as part of the IFC schema, has been followed.

The Pset_OpaqueMaterialProperties has been developed to capture the thermal properties of opaque materials of a structural building element. It includes seven properties with names "Roughness", "Conductivity", "Density", "SpecificHeat",

---

[17]https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/link/ifcmateriallayer

[18] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/ifcmaterialproperties

[19] https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/ifcpropertyset

"ThermalAbsorptance", "SolarAbsorptance" and "VisibleAbsorptance". Table 3 presents the template, type, and unit of each property.

Table 3 – Thermal properties of opaque materials: definition of the Pset_OpaqueMaterialProperties properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Type	Unit
Opaque Material Thermal Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - Material IFC: modelled as IfcPropertySets or IfcMaterialProperties	Pset_OpaqueMaterialProperties	Roughness	IfcProperty SingleValue	IfcText	-
		Conductivity	IfcProperty SingleValue	IfcThermalConductivityMeasure	W/mK
		Density	IfcProperty SingleValue	IfcMassDensityMeasure	kg/m^3
		SpecificHeat	IfcProperty SingleValue	IfcSpecificHeatCapacityMeasure	J/kg K
		ThermalAbsorptance	IfcProperty SingleValue	IfcReal	-
		SolarAbsorptance	IfcProperty SingleValue	IfcReal	-
		VisibleAbsorptance	IfcProperty SingleValue	IfcReal	-

Alternatively, the IfcMaterialProperties class could be used. The IfcMaterialProperties assigns a set of material properties to associated material definitions. Within BIMERR, the set is identified using the Pset_OpaqueMaterialProperties name. Although the property set approach, described above, is more straightforward, due to limitations of the xBIM (used by RenoDSS) on serialising and deserialising it, the IfcMaterialProperties-based approach is adopted. Hence, to generate objects of the Material class of EnergyPlus, the following rule is applied:

Rule 6: Each IfcMaterialLayer object is mapped to a Material object of EnergyPlus. To set the properties' values of each object, the values of the LayerThickness and the Material

attributes of the IfcMaterialLayer object are queried. The LayerThickness is directly mapped to the respective property of the Material class of EnergyPlus. The rest properties are set by querying the Properties of the Material attribute of the IfcMaterialLayer object.

With regards to the thermal properties of materials that are related with opening elements, the Pset_GlazingMaterialProperties class has been developed to capture three properties: (1) Ufactor; (2) SolarHeatGainCoefficient; and (3) VisibleTransmittance. Table 4 presents the template, type, and unit of each property.

Table 4 – Thermal properties of materials that refer to opening elements: definition of the Pset_GlazingMaterialProperties properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Type	Unit
Glazing Construction Thermal Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - WindowMaterial:SimpleGlazingSystem and Material:NoMass IFC: modelled as a Pset linked to each IfcWindowType or IfcDoorType	Pset_GlazingMaterialProperties	Ufactor	IfcProperty SingleValue	IfcThermalTransmitt anceMeasure	W/m^{2}K
		SolarHeatGai nCoefficient	IfcProperty SingleValue	IfcReal	-
		VisibleTrans mittance	IfcProperty SingleValue	IfcReal	-

To perform their mapping to objects of the WindowMaterial:SimpleGlazingSystem and Material:NoMass classes of EnergyPlus the following rule is applied:

Rule 7: For each IfcWindowType or IfcDoorType object, the value of its HasPropertySets attribute is queried, returning a list of IfcPropertySet objects. From that list, the IfcPropertySet with Name "Pset_GlazingMaterialProperties" is selected. Then, the value of

its IfcProperty with Name "SolarHeatGainCoefficient" is queried. If it equals zero a Material:NoMass element is created; otherwise, a WindowMaterial:SimpleGlazingSystem is created.

3.1.4 Energy Systems – HVAC supply and demand side components

Concerning the Energy Systems modelling and simulation, the IFC to IDF mapping has been limited to the actual HVAC system types occurred in the pre-validation and validation sites. For their modelling and simulation, the HVAC templates' group of EnergyPlus is utilised. Hence, combinations of HVACTemplate[20] objects that are currently supported by the BEP module are the following:

- For baseboard heating systems with (optional) hot water boiler, occurred in the Spanish and Polish validation and the CONKAT pre validation sites:
    - HVACTemplate:Thermostat;
    - HVACTemplate:Zone:BaseboardHeat; and
    - HVACTemplate:Plant:Boiler (optional).
- For variable refrigerant flow (VRF) heat pumps (air-to-air), occurred in the KRIPIS pre-validation site:
    - HVACTemplate:Thermostat;
    - HVACTemplate:Zone:VRF; and
    - HVACTemplate:System:VRF.
- For direct-expansion cooling, packaged and split system simulations, occurred in the CONKAT pre-validation site:
    - HVACTemplate:Thermostat;
    - HVACTemplate:Zone:Unitary; and
    - HVACTemplate:System:Unitary.

Due to lack of information related to the HVAC systems' modelling and limited experience of the BIM modellers on the HVAC data capturing in IFC, Revit families have been created to export the HVAC components and their property sets properly. The minimum

---

[20] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - Group-HVAC Templates

parameters that are required to perform the simulation of such components have been set as properties of that families. This approach makes the IFC to IDF mapping process less prone to human error, since predefined elements are translated automatically to HVAC components that are linked with certain energy related parameters. Additional parameters that are required for each system type's simulation are prefixed to "autosized". The "Autosized" term of EnergyPlus indicates its capability to automatically size many HVAC components (in other words, to auto-estimate the values of certain HVAC components' parameters) based on user-specified building configuration and specific sizing climate conditions (e.g., extreme summer and winter days).

For the HVACTemplate:Thermostat modelling, the Thermostat family has been designed. Each Thermostat element is exported as an IfcUnitaryControlElement of type "THERMOSTAT" linked with an IfcPropertySet, named Pset_Thermostat. The thermostat's Name, HeatingSetpoint and CoolingSetpoint comprise its properties set.

Rule 8: Each IfcUnitaryControlEment of type "THERMOSTAT" along with its Pset_Thermostat are mapped to an object of the HVACTemplate:Thermostat class of EnergyPlus.

Table 5 provides information about the name, type, value and unit of each property of the Pset_Thermostat.

Table 5 – Energy related properties of thermostats: definition of the Pset_Thermostat properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Type	Value	Unit
Thermostat Properties EnergyPlus: requirements defined	Pset_Thermostat	Name	IfcProperty SingleValue	IfcText	-
		HeatingSetpoint	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C

| | | CoolingSetpoint | IfcPropertySingleValue | IfcThermodynamicTemperatureMeasure | C |

| based on EnergyPlus 9.3 – HVACTemplate:Thermostat

IFC: modelled as IfcUnitaryControlElement (type: THERMOSTAT) | | CoolingSetpoint | IfcPropertySingleValue | IfcThermodynamicTemperatureMeasure | C |

Rule 9: Each IfcSpaceHeater object represents a radiator. It is mapped to an object of the HVACTemplate:BaseboardHeat class.

To capture data about its AvailabilitySchedule, HeatingCapacity and HeatingType, the Pset_RadiatorProperties has been developed. In Table 6, the Type, Value and Unit of each property of the Pset_RadiatorProperties properties set are defined.

Table 6 – Energy related properties of radiators: definition of the Pset_RadiatorProperties properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Type	Value	Unit
Space Heater - Radiator Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - HVACTemplate:BaseboardHeat IFC: modelled as IfcSpaceHeater	Pset_RadiatorProperties	AvailabilitySchedule	IfcPropertyReferenceValue	IfcIrregularTimeSeries	-
		HeatingCapacity	IfcPropertySingleValue	IfcPowerMeasure	W
		HeatingType	IfcPropertySingleValue	IfcLabel (HotWater, Electric)	-

For the Boilers modelling, the Boiler family in Revit has been designed. Its properties have been properly set to allow exportation of each Boiler as an IfcBoiler object linked with an IfcPropertySet, named Pset_BoilerProperties. Among a wide range of properties that are required to simulate Boilers, the BoilerType, Capacity, Efficiency, MinimumPartLoadRatio,

MaximumPartLoadRatio, OptimumPartLoadRatio and WaterOutletUpperTemperature Limit comprise the properties set of the Pset_BoilerProperties.

Rule 10: Each IfcBoiler object along with its Pset_BoilerProperties are mapped to an object of the HVACTemplate:Boiler class of EnergyPlus.

Table 7 provides information about the name, type, value and unit of each property of the Pset_BoilerProperties.

Table 7 – Energy related properties of boilers: definition of the Pset_BoilerProperties properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
Boiler Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - HVACTemplate:Boiler IFC: modelled as IfcBoiler	Pset_BoilerProperties	BoilerType	IfcProperty SingleValue	IfcLabel (DistrictHotWater, HotWaterBoiler, CondensingHotWaterBoiler)	-
		Capacity	IfcProperty SingleValue	IfcPowerMeasure	W
		Efficiency	IfcProperty SingleValue	IfcReal	-
		FuelType	IfcProperty SingleValue	IfcLabel (Electricity, NaturalGas, Diesel)	-
		MinimumPartLoadRatio	IfcProperty SingleValue	IfcRatioMeasure	-
		MaximumPartLoadRatio	IfcProperty SingleValue	IfcRatioMeasure	-
		OptimumPartLoadRatio	IfcProperty SingleValue	IfcRatioMeasure	-
		WaterOutletUpperTemperatureLimit	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C

A VRF system consists of supply side and demand side components. In terms of EnergyPlus, HVACTemplate:Zone:VRF and HVACTemplate:System:VRF objects are used to simulate the demand and supply side components of the VRF system, respectively. To capture their input data requirements in IFC, two Revit families have been designed. The supply and demand side components are exported as IfcUnitaryEquipment objects of type "AIRCONDITIONINGUNIT" and "ROOFTOPUNIT", respectively. Each IfcUnitary Equipment object of type "AIRCONDITIONINGUNIT" object is linked with an IfcPropertySet with Name Pset_VRFSupplyProperties that includes all the properties required for its simulation. In a similar way, each IfcUnitaryEquipment object of type "ROOFTOPUNIT" object is linked with an IfcPropertySet with name Pset_VRFDemandProperties that includes all the properties required for its simulation. Further information about these properties' sets is provided in Table 8.

Rule 11, 12: In the IFC to IDF mapping process, (rule 11) each IfcUnitaryEquipment object of type "AIRCONDITIONINGUNIT", along with its properties set, is mapped to a HVACTemplate:System:VRF object, while (rule 12) each IfcUnitaryEquipment object of type "ROOFTOPUNIT", along with its properties set, is mapped to a HVACTemplate:Zone:VRF object.

Table 8 – Energy related properties of VRF systems (supply and demand side): definition of the Pset_VRFSupplyProperties and Pset_VRFDemandProperties properties sets.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
VRF supply side Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - HVACTemplate:System: VRF	Pset_VRFSupplyProperties	Name	IfcProperty SingleValue	IfcText	-
		EER	IfcProperty SingleValue	IfcReal	-
		COP	IfcProperty SingleValue	IfcReal	-
		Consumptio nCoolingCap acity	IfcProperty SingleValue	IfcPowerMeasure	W

IFC: modelled as IfcUnitaryEquipment (type: AIRCONDITIONINGUNIT)		ConsumptionHeatingCapacity	IfcProperty SingleValue	IfcPowerMeasure	W
		EfficiencyCoolingCapacity	IfcProperty SingleValue	IfcPowerMeasure	W
		EfficiencyHeatingCapacity	IfcProperty SingleValue	IfcPowerMeasure	W
		MaxOutdoor Temperature Cooling	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		MinOutdoor Temperature Cooling	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		MaxOutdoor Temperature Heating	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		MinOutdoor Temperature Heating	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
VRF demand side Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - HVACTemplate:Zone:VRF IFC: modelled as IfcUnitaryEquipment (type: ROOFTOPUNIT)	Pset_VRFDemandProperties	SupplySideSystemName	IfcProperty SingleValue	IfcText	-
		CoolingCapacity	IfcProperty SingleValue	IfcPowerMeasure	W
		HeatingCapacity	IfcProperty SingleValue	IfcPowerMeasure	W

With regards to the residential AC units, the ResidentialAC family has been designed, encompassing the minimum properties set that is required to perform an AC unit

simulation. Each ResidentialAC object is exported as an IfcUnitaryEquipment object of type "SPLITUNIT" accompanied by an IfcPropertySet with Name Pset_ResidentialACunits Properties.

Rule 13: The IFC to IDF mapping process transforms each IfcUnitaryEquipment object of type "SPLITUNIT" along with its Pset_ResidentialACunitsProperties to a combination of a HVACTemplate:Zone:Unitary and HVACTemplate:System:Unitary objects.

Table 9 – Energy related properties of residential AC units: definition of Pset_ResidentialACunits.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
Residential AC units Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - HVACTemplate:Zone:Unitary and HVACTemplate:System:Unitary IFC: modelled as IfcUnitaryEquipment (type: SPLITUNIT)	Pset_ResidentialACunitsProperties	ConsumptionCoolingCapacity	IfcPropertySingleValue	IfcPowerMeasure	W
		ConsumptionHeatingCapacity	IfcPropertySingleValue	IfcPowerMeasure	W
		EER	IfcPropertySingleValue	IfcReal	-
		COP	IfcPropertySingleValue	IfcReal	-
		CoolingDesignSupplyAirTemperature	IfcPropertySingleValue	IfcThermodynamicTemperatureMeasure	C
		HeatingDesignSupplyAirTemperature	IfcPropertySingleValue	IfcThermodynamicTemperatureMeasure	C
		SupplyFanTotalEfficiency	IfcPropertySingleValue	IfcReal	-
		SupplyFanDeltaPressure	IfcPropertySingleValue	IfcPressureMeasure	Pa

		SupplyFanM otorEfficienc y	IfcProperty SingleValue	IfcReal	-

3.1.5 Internal Gains and Renewables

Heat gains from lights (artificial lighting), people, and electric equipment at each space (thermal zone) appertain to the internal gains' category. To support the respective internal gains' simulation, EnergyPlus provides the Lights, People and ElectricEquipment classes[21].

Although IFC schema supports the representation of internal gains scheduling data (IfcTimeSeries), it cannot capture information about their nominal values and other parameters that are required for their simulation. To overcome this limitation an IfcPropertySet, that has been developed in the context of OptEEmAL project[22], is adopted, named Pset_SpaceInternalGainsDesign, and linked with each IfcSpace. Table 10 presents this property set.

Rule 14: In the IFC to IDF mapping process, for each Pset_SpaceInternalGainsDesign, connected with an IfcSpace, an object of the following classes of EnergyPlus is populated: People; Lights; ElectricEquipment; and ZoneInfiltration:DesignFlowRate. The ZoneName of each object is set to the IfcSpace name that the Pset_SpaceInternalGainsDesign is connected with, whereas values of other properties are mapped effortlessly.

Table 10 – Energy related properties of Internal Gains: definition of the Pset_SpaceInternalGains properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit

---

[21] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - Group-Internal Gains

[22] https://www.opteemal-project.eu/

		OccupancySchedule	IfcPropertyReferenceValue	IfcIrregularTimeSeries	-
Internal Gains Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - Internal Gains	Pset_SpaceInternalGains	LightingSchedule	IfcPropertyReferenceValue	IfcIrregularTimeSeries	-
		EquipmentSchedule	IfcPropertyReferenceValue	IfcIrregularTimeSeries	-
		NumberOfOccupants	IfcPropertySingleValue	IfcInteger	-
		HeatGainPerOccupant	IfcPropertySingleValue	IfcPowerMeasure	W
		HeatGainLighting	IfcPropertySingleValue	IfcPowerMeasure	W
		HeatGainEquipment	IfcPropertySingleValue	IfcPowerMeasure	W
		InfiltrationRate	IfcPropertySingleValue	IfcVolumetricFlowRateMeasure	m^3/s
		Occupiable	IfcPropertySingleValue	IfcBoolean	-

Taking into account the characteristics of actual renewables systems in the KRIPIS pre-validation site and the requirements for energy performance evaluation of renewables renovation measures, within BIMERR, two types of renewables systems' modelling and simulation are supported: (1) solar collectors connected with hot water tanks; and (2) photovoltaic systems.

Concerning their modelling in EnergyPlus, when it comes to the solar collectors connected with hot water tanks simulation, the Solar Collectors and Water Heaters groups are considered. To capture relevant information in IFC, two dedicated Revit families have been created that allow exportation of IfcSolarDevice objects of type "SOLARCOLLECTOR" and IfcTank objects of type "STORAGE", linked with IfcPropertySet objects with name

Pset_SolarCollectorProperties and Pset_WaterHeaterStorageProperties, respectively. Further information is provided in Table 11 and Table 12.

Table 11 – Energy related properties of Solar Collectors: definition of the Pset_SolarCollectors properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
Solar Collectors Properties EnergyPlus: Requirements defined based on EnergyPlus 9.3 - Solar Collectors IFC: modelled as IfcSolarDevice (type: SOLARCOLLECTOR)	Pset_SolarCollector Properties	MaximumFlowRate	IfcProperty SingleValue	IfcVolumetricFlowRa teMeasure	m^3/s
		SurfaceName	IfcProperty SingleValue	IfcText	
		GrossArea	IfcProperty SingleValue	IfcAreaMeasure	m^2
		TestFlowRate	IfcProperty SingleValue	IfcVolumetricFlowRa teMeasure	m^3/s
		Coefficient1ofEfficiencyEquation	IfcProperty SingleValue	IfcReal	-
		Coefficient2ofEfficiencyEquation	IfcProperty SingleValue	IfcReal	-
		Coefficient3ofEfficiencyEquation	IfcProperty SingleValue	IfcReal	-
		Coefficient2ofIncidentAngleModifier	IfcProperty SingleValue	IfcReal	-
		Coefficient3ofIncidentAngleModifier	IfcProperty SingleValue	IfcReal	-

Rule 15, 16: In the IFC to IDF mapping process, each IfcSolarDevice object of type "SOLARCOLLECTOR", in combination with its Pset_SolarCollectorProperties, is mapped to

a SolarCollector:FlatPlate:Water object and a SolarCollectorPerformance:FlatPlate object of EnergyPlus (rule 15). Concerning the IfcTank objects of type "STORAGE" and their properties provided by the respective Pset_WaterHeaterStorageProperties, they are transformed to a WaterHeater:Mixed object (rule 16).

Table 12 – Energy related properties of Hot Water Tanks: definition of the Pset_WaterHeaterStorage properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
Water Heater/Storage Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - Water Heaters IFC: modelled as IfcTank (type: STORAGE)	Pset_WaterHeaterStorageRroperties	TankVolume	IfcProperty SingleValue	IfcVolumeMeasure	m^3
		HotWaterSetpointTemperature	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		DeadbandTemperatureDifference	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		MaximumTemperatureLimit	IfcProperty SingleValue	IfcThermodynamicTemperatureMeasure	C
		HeaterMaximumCapacity	IfcProperty SingleValue	IfcPowerMeasure	W
		HeaterFuelType	IfcProperty SingleValue	IfcLabel (Electricity, NaturalGas, Diesel, DistrictHeating)	-
		HeaterThermalEfficiency	IfcProperty SingleValue	IfcReal	-
		UseSideDesignFlowRate	IfcProperty SingleValue	IfcVolumetricFlowRateMeasure	m^3/s

		SourceSideD esignFlowRat e	IfcProperty SingleValue	IfcVolumetricFlowRa teMeasure	m³/s

As far as the photovoltaic systems' modelling and simulation are concerned, the Photovoltaic Generators[23] group of EnergyPlus is used. Similar to the solar collectors' system, to capture relevant information in IFC, a Revit family has been created that allow exportation of IfcSolarDevice objects of type "SOLARPANEL", linked with IfcPropertySet objects with name Pset_PhotovoltaicsProperties. Further information about this property set is provided in Table 13.

Rule 17: Each IfcSolarDevice object of type "SOLARPANEL" along with its properties captured by the property set with name Pset_PhotovoltaicsProperties is treated as a combination of a Generator:Photovoltaic and PhotovoltaicPerforamance:Simple objects of EnergyPlus.

Table 13 – Energy related properties of Photovoltaics: definition of the Pset_Photovoltaics properties set.

EnergyPlus Requirements	PsetName	Properties			
		Property Name	Template	Value	Unit
Photovoltaics Properties EnergyPlus: requirements defined based on EnergyPlus 9.3 - Photovoltaic Generators)	Pset_Photovoltaics Properties	SurfaceName	IfcProperty SingleValue	IfcText	
		RatedElectric PowerOutput	IfcProperty SingleValue	IfcPowerMeasure	W
		NumberOfSe riesStringsIn Parallel	IfcProperty SingleValue	IfcCountMeasure	-

---

[23] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - Group - Photovoltaic Generators

IFC: modelled as IfcSolarDevice (type: SOLARPANEL)		NumberOfModulesinSeries	IfcPropertySingleValue	IfcCountMeasure	-
		FractionOfSurfaceWithActiveSolarCells	IfcPropertySingleValue	IfcRatioMeasure	-
		CellEfficiency	IfcPropertySingleValue	IfcReal	-

3.1.6   Other Simulation Parameters

An efficient selection of (a) surface convection, (b) conduction and heat balance algorithm, (c) equipment and system sizing, (d) daylighting, (e) dynamic fenestration controls airflow analysis model, along with (f) simulation start and end time (defined by: month, day, hour and minute), (g) simulation inter-sample time interval (in minutes), (h) preferred outputs, to name but a few, requires domain expertise that none of the current BIMERR data models can capture. These features fall under the other simulation parameters category and influence the simulation performance in various ways.

The inclusion of intermediate data models (e.g. SimModel [28]) as part of the BIMERR data models could be investigated towards bridging the "Other Simulation Parameters" data gap between BIM and BEPs. However, the added value of that inclusion would be minimal: (1) none other component of the BIMERR toolchain requires such data; (2) none of the BIMERR data providers (BIM Digital model creation tools) to the BIMERR Interoperability Framework (BIF) is capable to generate the "Other Simulation Parameters" information.

Rule 18: To meet the "Other Simulation Parameters"[24] data requirements, the EnergyPlus IDF is enriched with one instance of each of the following classes: SimulationControl; Building; RunPeriod; Timestep; SurfaceConvectionAlgorithm:Inside; SurfaceConvection Algorithm: Ouside; HeatBalanceAlgorithm, whose properties have been carefully set to

---

prefixed values, according to several simulations that run and were analysed to find the combination that keeps the simulation accuracy and runtime in balance.

3.1.7   Time-variant Data – Occupant Behaviour Models

To achieve a modular-like co-simulation between obXML and EnergyPlus, the FMUs-based approach has been selected.

Rule 19: To prepare EnergyPlus for such a data exchange, instances of the following IDD classes are automatically generated, based on the obXML content:

- ExternalInterface:FunctionalMockupUnitExport:From:Variable[25]; and
- ExternalInterface:FunctionalMockupUnitExport:To:Schedule[26].

Instances of the former class are used to expose the EnergyPlus outputs to obXML (e.g zone temperature, relative humidity), while instances of the latter are used to set the time-variant data of different controllable elements in EnergyPlus (e.g. thermostat setpoints, occupant presence) to the values retrieved from the obXML simulation.

To reduce the total simulation runtime, including the time that is required by the IFC to IDF mapping and Co-Simulation setup processes, in the second release of BEP an additional approach is being supported. It integrates the EnergyPlus Python Data Exchange API[27].

Rule 20: To support the data exchange with obXML, an object of the PythonPlugin: Instance [28] class is generated. This object captures a single plugin instance for EnergyPlus, that simulates how occupants operate the HVAC components of the building.

---

[25] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - FMU export from Variable

[26] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - FMU export to Schedule

[27] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - Data Exchange API

[28] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - PythnonPlugin:Instance

3.2 ENERGYPLUS IDF AND OBXML CO-SIMULATION

Concerning the FMUs-based approach, having completed the automated EnergyPlus IDF generation, both EnergyPlus IDF and obXML are automatically exported to FMUs, utilizing dedicated software packages. EnergyPlusToFMU[29], a software package written in Python, is triggered to generate the FMU using EnergyPlus IDF and EPW files as inputs, and obFMU[30] receives the obXML as input and generated the relevant FMU. Both FMUs are then imported to pyFMI to perform the co-simulation.

In the approach that integrates the EnergyPlus Python Data Exchange API, the obXML data are being processed to automatically generate a python script. This script interacts with EnergyPlus in the "Plugin Mode". The following functions of the Data Exchange API are used for a seamless data exchange between EnergyPlus and the auto-generated python script:

- set_actuator_value, to set the value of an actuator in a running simulation, estimated by the respective occupant behaviour model;
- get_variable_value, to get the current value of a variable in a running simulation that acts as input for the occupant behaviour model.

3.3 SIMULATION OUTPUT – POST PROCESSING

Output of the aforementioned co-simulation, regardless the approach that is followed for the co-simulation setup, is a CSV file, which contains reported values of output variables that have been defined by instances of IDD's Output:Variable class. At this point it is worth mentioning that, by default, the variable names are prefixed, according to the name-coding that the EnergyPlus back-end understands to perform the calculations. However, EnergyPlus provides to the user the flexibility to report the same variables with custom

---

[29] FMU Export of EnergyPlus - EnergyPlusToFMU

[30] OB Modeling Tool - obFMU

variable names, utilizing the EnergyManagementSystem:Sensor[31] class. The BEP manager is responsible for translating the output-CSV to a meaningful JSON format file that the RenoDSS data management module expects. In order to facilitate this post processing method of the BEP module, instances of the EnergyManagementSystem:Sensor class are generated, whose names are set to the KPI IDs that are listed in Section 2.1.

3.4 TECHNOLOGY STACK AND IMPLEMENTATION TOOLS

According to the BIMERR architecture, the BEP module acts as the RenoDSS backend for the Energy KPIs calculation. Within T7.5, where the RenoDSS UI development and its components integration are considered, state-of-the-art technologies (e.g. Angular, Typescript, NGRX Entity/Store, Leaflet, and PostgreSQL) are utilised for the presentation (frontend) and the data storage layers, as they have been documented in the D7.10 – *Integrated Renovation Decision Support System 2* [29].

Concerning the business logic layer, the second release of the BEP module has been developed in Python 3.7.6, utilising backend technologies that are listed in Table 14.

Table 14: Technologies and libraries used in BIMERR BEP module

Name of the technology/library	Version	License
EnergyPlus	9.3	BSD 3-Clause
ObXML	1.3.3	BSD 3-Clause
ObFMU	1.3.3	BSD 3-Clause
EnergyPlusToFMU	3.0.0	BSD 3-Clause
pyFMI	2.5	LGPL-3.0 License
IfcOpenShell	0.5.0	LGPL-3.0 License
Flask	1.1.1	BSD 3-Clause
Flask Restful	0.3.8	BSD 3-Clause
Spring MVC	4.3.5	Apache License 2.0 license

EnergyPlus is the simulation engine that performs the Energy KPIs calculation (for further information refer to Section 2.3). IfcOpenShell is an open source (LGPL-3.0 License) library,

---

[31] bigladdersoftware - EnergyPlus 9.3 - I/O Reference - EMS:Sensor

available in Python, for working with the Industry Foundation Classes (IFC) file format. ObFMU is a tool that, based on the obXML, creates the occupant behaviour Functional Mock-up Unit (FMU) for co-simulation using the Functional Mock-up Interface (FMI). Eppy is a python library for EnergyPlus IDF files and EnergyPlus output files handling, providing functions for navigation, search, and modification of the IDF instances. EnergyPlusToFMU is a software package written in Python used for exporting the building simulation program EnergyPlus as a Functional Mock-up Unit (FMU) for co-simulation. PyFMI is a python FMI library for loading and interacting with Functional Mock-Up Units (FMUs), which are compiled dynamic models compliant with the Functional Mock-Up Interface (FMI). Flask micro web framework and its Restful extension are used to expose the BEP module as a service.

BEP has been developed as a web application using the Spring MVC framework. The Spring Web MVC framework provides a Model-View-Controller (MVC) architecture and ready components that can be used to develop flexible and loosely coupled web applications. Spring MVC also offers the flexibility for building quickly REST web interfaces for communicating with 3rd party applications. REST embraces the precepts of the web, including its architecture and its benefits.
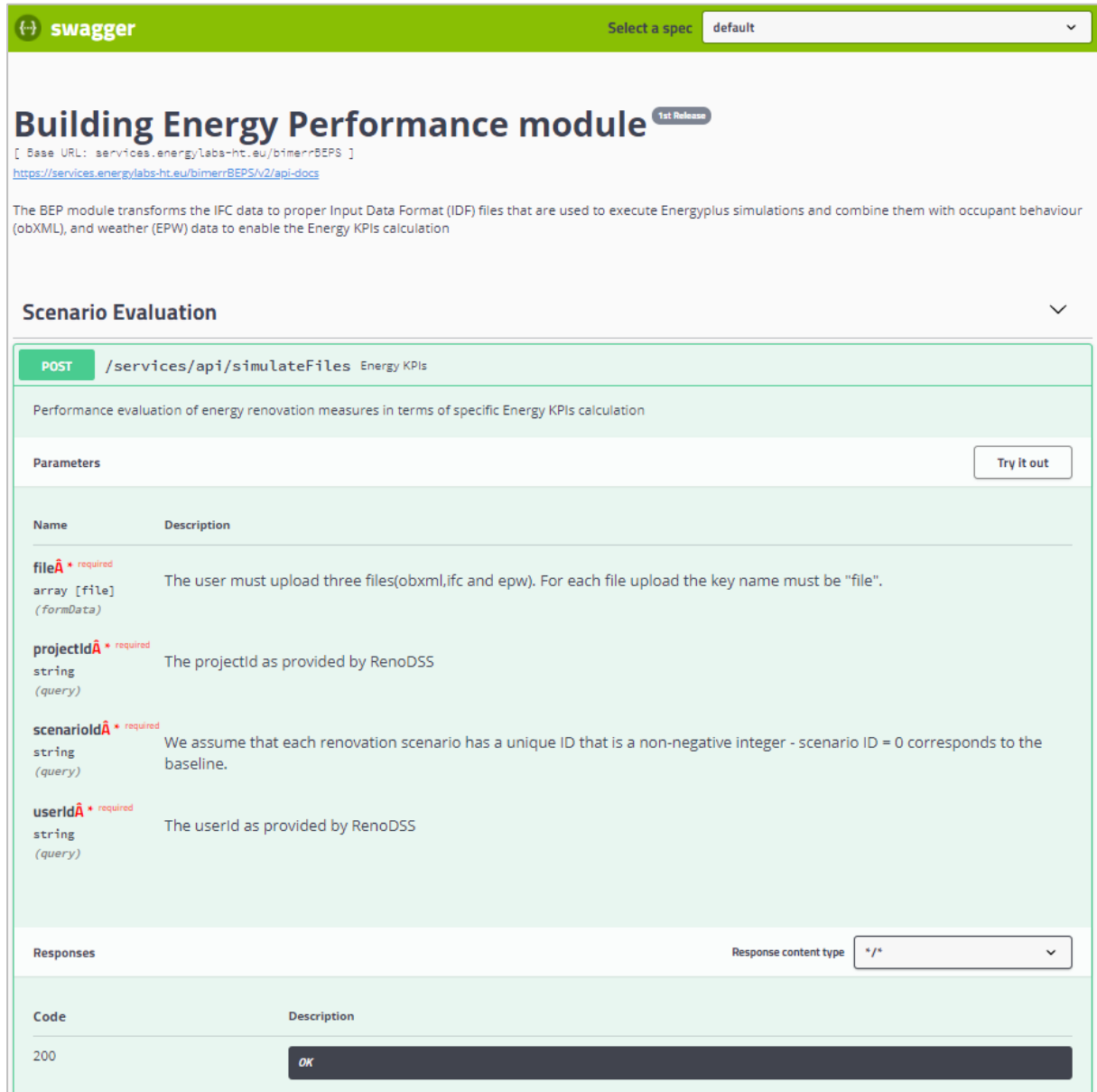
BEP offers asynchronous and parallel processing in all of its REST request using standard asynchronous libraries from the Spring MVC framework. That means that multiple concurrent requests could be served at the same time facilitating and optimizing the communication with RenoDSS.

3.5   API DOCUMENTATION

The data exchange between the RenoDSS data management and the BEP module is achieved as follows:

- IFC, obXML and EPW files, scenario ID, user ID and project ID are sent from RenoDSS data management module to the BEP module via HTTP Post requests to the BEP REST endpoint.

- The BEP module backend data pre-processing, transformation, co-simulation, and post-processing methods are performed to generate an Energy KPIs JSON file, which constitutes the response of the request.

Figure 7 The BEP module API documented in Swagger[32]

---

[32] https://services.energylabs-ht.eu//bimerrBEPS/swagger-ui.html

In summary, the BEP module receives as input three different types of files (.epw .ifc .xml) and three String parameters, the userId, the projectId and the scenarioID. The BEP module ensures that the end user has selected three files (neither less nor more) of the aforementioned file types.

Asynchronous communication between the RenoDSS data management system and the BEP module is supported. The BEP module can handle many requests for different scenarios energy KPIs calculation. When, the simulation of a scenario is completed, the results are forwarded to the RenoDSS data management system by sending a post request to the following endpoint, adding an "apiKey" header for security reasons:

- https://renodss.xylem-technologies.com/api/m2m/bep/kpis

The BEP module API is documented in Swagger as Figure 7 depicts.

3.6 ASSUMPTIONS AND RESTRICTIONS

Prerequisites for the successful energy simulation of the baseline and potential renovation scenarios are the following:

- Static data – Valid IFC file containing: (1) second-level space boundaries; (2) materials' thermal properties; (3) internal gains; (4) HVAC components; (5) renewables (optional) and their energy related properties populated in alignment with the properties sets that are described in Section 3.1;
- Dynamic data – Valid ObXML file containing occupant behaviour data for each space, generated by the PRUBS component, and receiving occupants' feedback using the Building Information Collection Application;
- Weather data – Valid EnergyPlus weather data file (EPW) containing weather information for a typical meteorological year; extreme summer and winter days must be included.
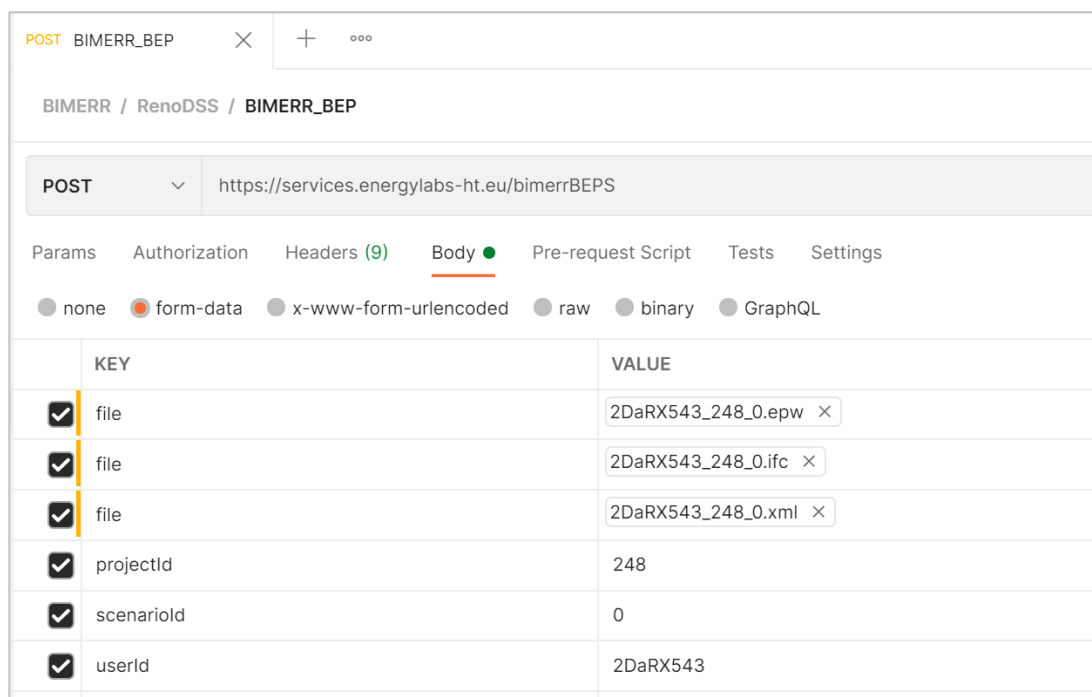
3.7 INSTALLATION INSTRUCTIONS

The BIMERR BEP module is accessible via a rest API, thus installation or downloading of any component is not required.

3.8 APPLICATION EXAMPLES

As an initial demonstration of the BEP module performance, a hypothetical building whose IFC file is enriched with the 2nd-level space boundaries [27], internal gains, and material thermal properties, has been selected. It is a two-storey building, located in Athens - Greece, consisting of six spaces and a variety of components including transparent and opaque doors, windows, and curtain walls which also contain opaque and transparent sub-components (doors, windows and plates).

As Figure 9 illustrates, the EnergyPlus Simulation Input Data File Generation tool correctly identifies: (1) Building and Fenestration surfaces; (2) their boundary condition type (different colours); (3) the corresponding surfaces, when boundary condition type is surface"; (4) the surface type; (5) the surface construction; and (5) the 3D geometry representation of each surface.

Figure 8 – Postman request to the BEP module endpoint for the baseline Energy KPI calculation
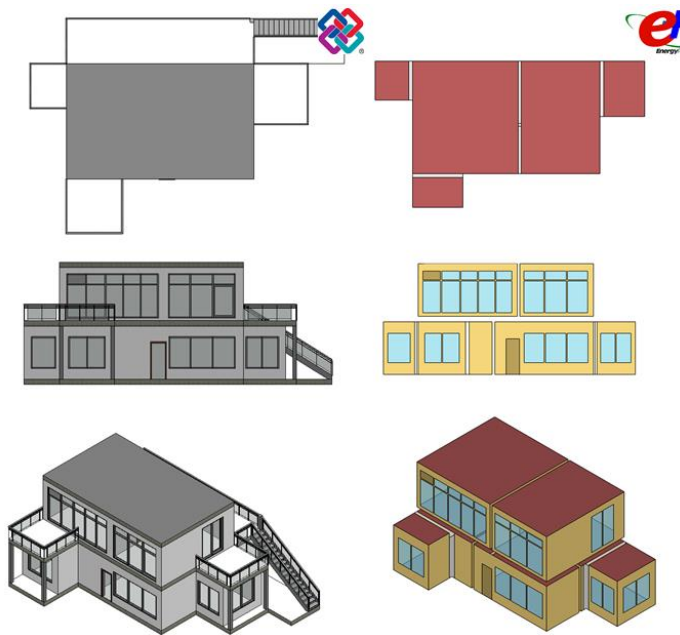
Figure 9 – IDF Generator results: Rendering the generated IDF geometry in SketchUp

Moreover, Figure 10 shows that the IDF Generator manages to map information of materials, thermal properties, and construction of the building elements properly.
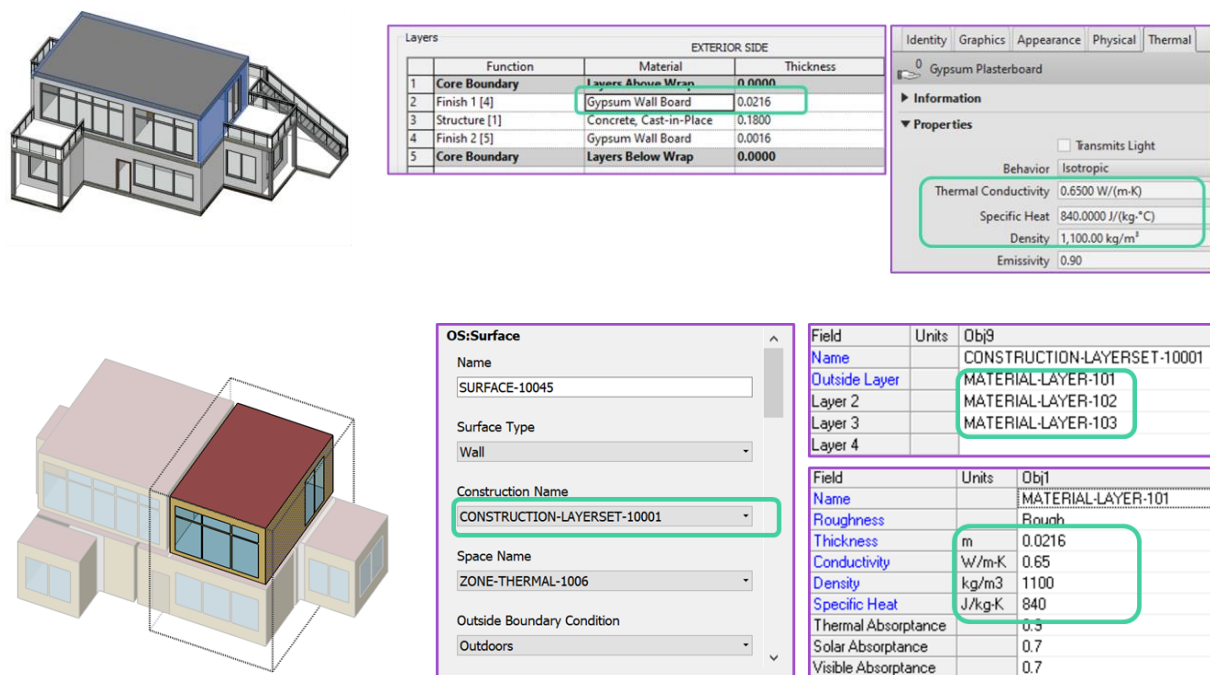

Figure 10 – IDF Generator results: Construction (material layers sets) and material thermal properties
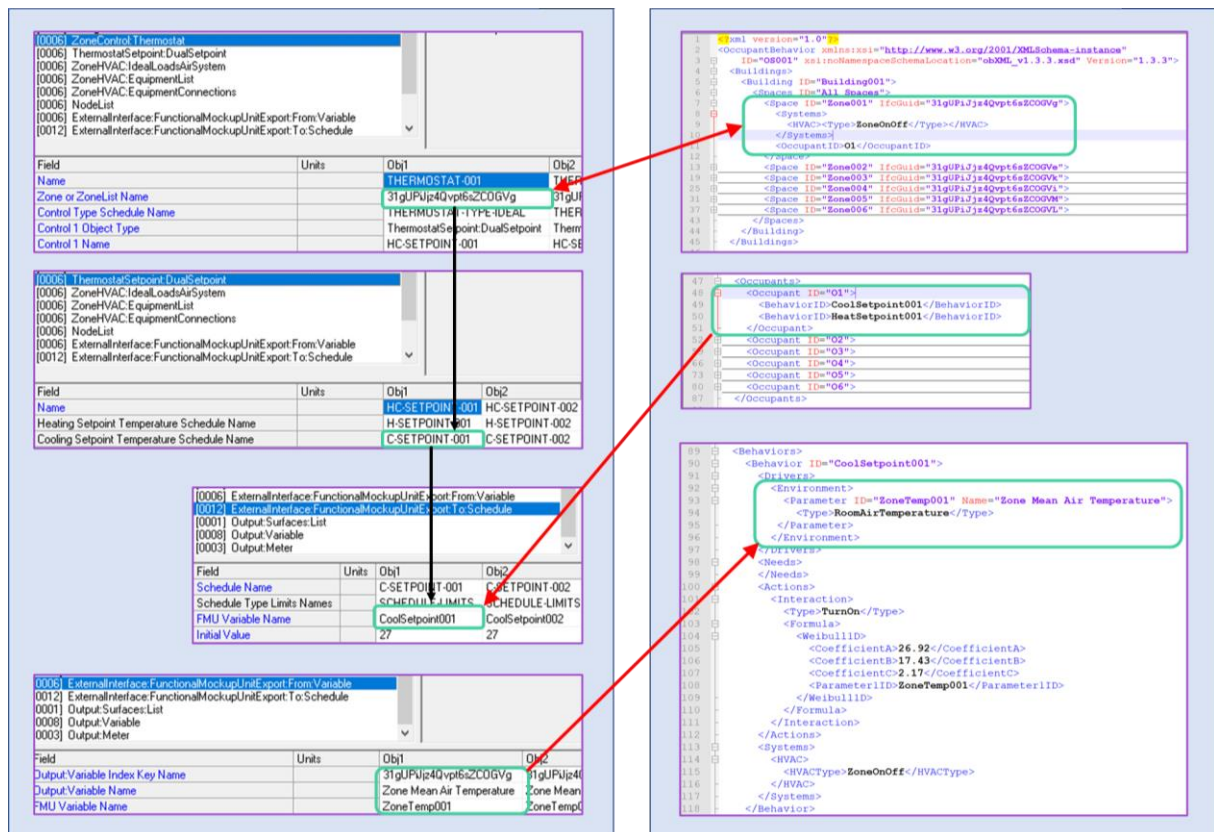
Figure 11 – EnergyPlus and ObXML co-simulation: parameters setup: EnergyPlus sends the zone air temperature to and receives the thermostat cooling setpoint from the obXML at each timestep.

With the IDF correctly generated, ObXML and EnergyPlus are co-simulated to eventually calculate the Energy KPIs. A subset of the co-simulated parameters setup is depicted in Figure 11.

The EnergyPlus Output file includes all the data requested and post-processed to populate the Energy KPIs JSON data model, which constitutes the response to the previously described request (see Figure 8). A subset of that response is presented in Figure 12. Note here that the year values are missing from the timestamps of the Energy KPIs JSON, since the reported KPIs values are for a typical, and not specific, annual meteorological year. In more detail, there are two main different categories of climate data sources: (1) Annual Meteorological Year (AMY) that includes measured data from a local weather station and for a specific year; (2) Typical Meteorological Year (and more precisely TMY3) that contains representative climate data for a given location - here data for the last 15 years measured from weather stations located in airports are collected and statistically process to conclude to typical/representative weather data for each hour of

the year. Although TMY3 files cannot capture local microclimate effects, research studies in the BEPs domain have concluded that AMYs are not appropriate for annual BEP simulation and a decision-making process (evaluation on candidate renovation measures), since a single year AMY is not representative of a longer period. For the specific experiment, a TMY-source EPW file has been used, hence the year values are missing.

```
{
    "userID": "2DaRX543",
    "projectID": "248",
    "scenarioID": "0",
    "EnergyKPIs": [
        {
            "KPI_Identifier": "EN1",
            "KPI_Name": "Total primary energy consumption",
            "KPI_Units": "kWh.m^2.year",
            "KPI_Values": [
                {
                    "Value": "65.2",
                    "Timestamp": "12/31 24:00:00"
                }
            ]
        },
        {
            "KPI_Identifier": "EN2",
            "KPI_Name": "PENRT Primary energy non-renewable total",
            "KPI_Units": "kWh.m^2.year",
            "KPI_Values": [
                {
                    "Value": "65.2",
                    "Timestamp": "12/31 24:00:00"
                }
            ]
        },
        {
            "KPI_Identifier": "EN3",
            "KPI_Name": "Electric energy consumption",
            "KPI_Units": "kWh.m^2.year",
            "KPI_Values": [
                {
                    "Value": "65.2",
                    "Timestamp": "12/31 24:00:00"
                }
            ]
        },
        {
            "KPI_Identifier": "EN4",
            "KPI_Name": "Natural gas energy consumption",
            "KPI_Units": "kWh.m^2.year",
            "KPI_Values": [
                {
                    "Value": "0.0",
                    "Timestamp": "12/31 24:00:00"
                }
            ]
        },
```

```
{
    "KPI_Identifier": "EN8",
    "KPI_Name": "Heating load profile",
    "KPI_Units": "W",
    "KPI_Values": [
        {
            "Value": "3200.0",
            "Timestamp": "01/31 24:00:00"
        },
        {
            "Value": "4650.0",
            "Timestamp": "02/28 24:00:00"
        },
        {
            "Value": "3320.0",
            "Timestamp": "03/31 24:00:00"
        },
        {
            "Value": "2100.0",
            "Timestamp": "04/30 24:00:00"
        },
        {
            "Value": "670.0",
            "Timestamp": "05/31 24:00:00"
        },
        {
            "Value": "0.0",
            "Timestamp": "06/30 24:00:00"
        },
        {
            "Value": "0.0",
            "Timestamp": "07/31 24:00:00"
        },
        {
            "Value": "0.0",
            "Timestamp": "08/31 24:00:00"
        },
        {
            "Value": "0.0",
            "Timestamp": "09/30 24:00:00"
        },
        {
            "Value": "450.0",
            "Timestamp": "10/31 24:00:00"
        },
```

Figure 12 Subset of the Energy KPIs JSON – the BEP module response

The BEP module has also been demonstrated in the pre-validation pilot sites. As input, the BIM models, designed in Revit and including Revit families that have been designed for the HVAC and the renewables modelling, have been exported as IFC files using the modified IFC exporter that have been developed within BIMERR [10].
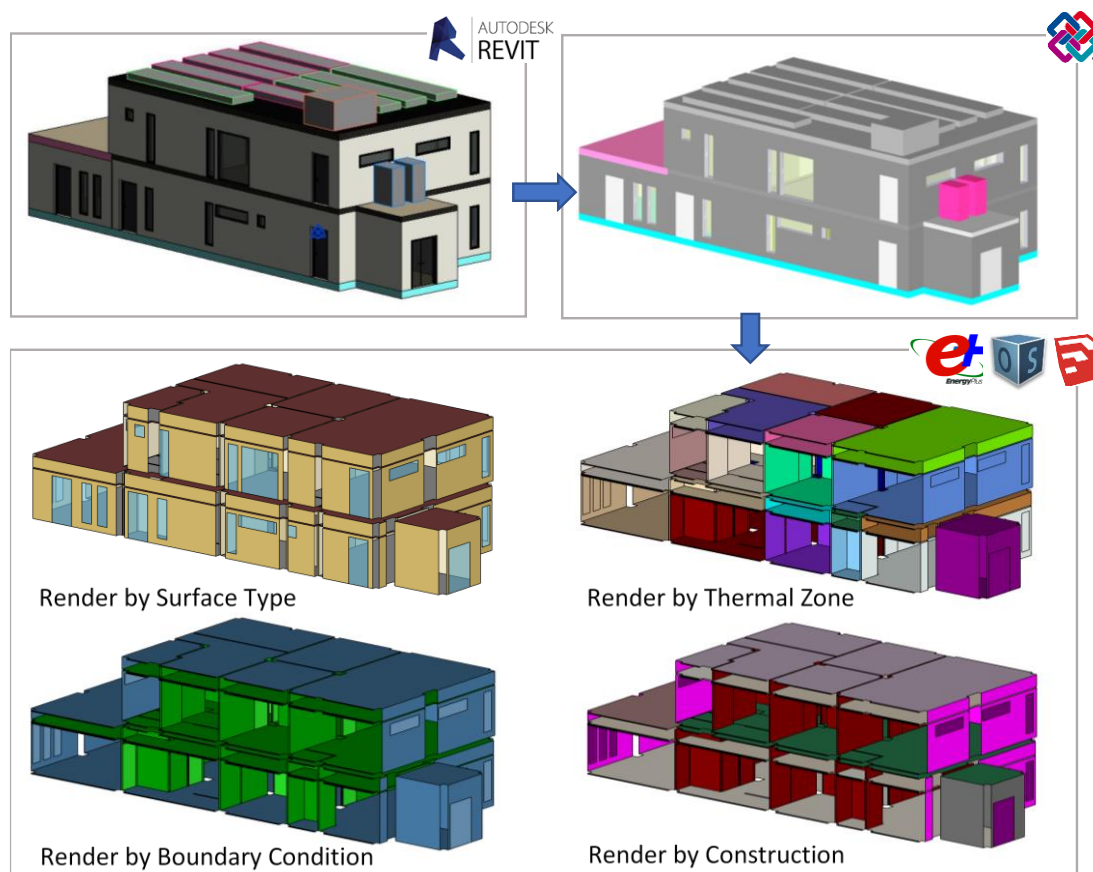
Figure 13 – KRIPIS Building: IDF Generator results – Rendering the generated IDF geometry in SketchUp.

Concerning the automated IFC to IDF mapping process, performed by the IDF Generator sub-component of BEP, the results of the thermal zoning, geometric representation, materials and construction mapping rules for the KRIPIS and CONKAT buildings are depicted in Figure 13 and Figure 14, respectively. The BEP module's mapping process correctly identifies: (1) surfaces' types; (2) their boundary condition type; (3) their construction & materials; and (4) thermal zones. The results have been rendered using the OpenStudio plugin for SketchUp[33]. Further results of the BEP module's demonstration in BIMERR pre-validation sites will be documented in D8.6 – *Report on BIMERR pre-validation activities* (to be submitted in September, 2021).

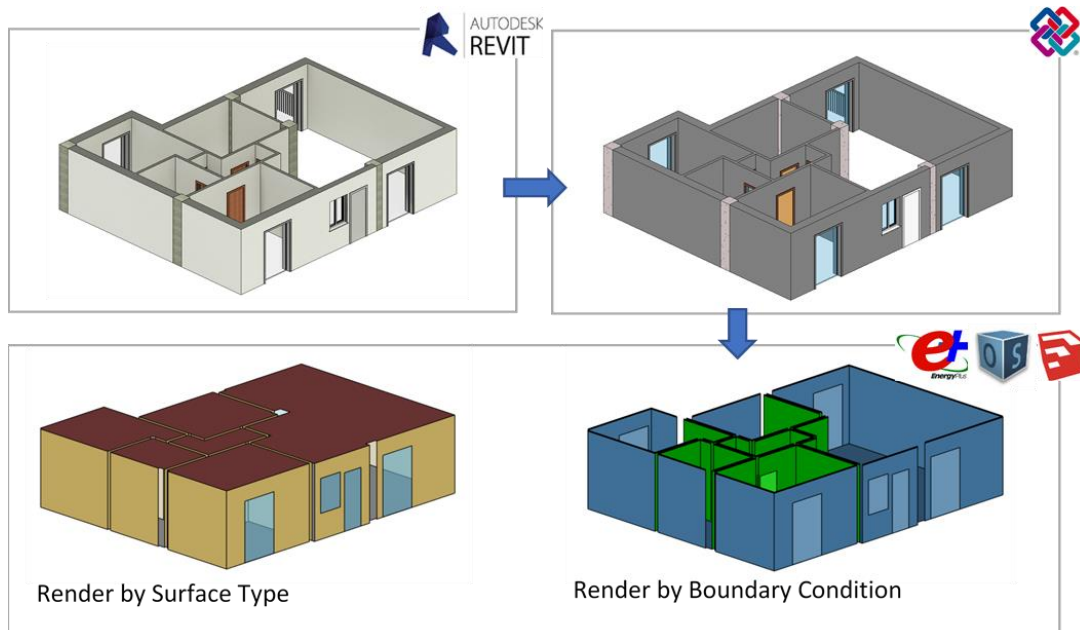
[33] https://www.openstudio.net/downloads

Figure 14 – CONKAT Building: IDF Generator results – Rendering the generated IDF geometry in SketchUp.

3.9   LICENSING

The BIMERR BEP module is a closed source component.

4. CONCLUSIONS

The focus of this deliverable has been the documentation of Building Energy Performance module; a module that (1) retrieves information for the RenoDSS data management module; (2) transforms that information to proper simulation input data files; (3) launches appropriate simulation engines and (4) aggregates simulation outputs to compute predefined Energy KPIs and communicate them back to RenoDSS.

In alignment with T7.3 work-planning, a survey on existing BEP simulation methods and the selection of the BEP simulation engine that best fits the project objectives have been completed, while the final version of functionalities, data flows specifications, interfaces and messages have been delivered as planned.

Since this work reflects the activities that have been accomplished for the second release of the module, assumptions and restriction that had been identified in the first release of the module have been addressed. Additional rules and functionalities that deal with the HVAC and renewables data transformation and supplementary data transfer requirements for candidate renovation scenarios evaluation have been implemented. Furthermore, towards decreasing the BEP module's response time and taking into account that several renovation scenarios evaluation requests will occur, asynchronous communication is now supported. On top of that, since the FMUs-based co-simulation setup significantly increases the simulation runtime, an alternative approach for the obXML and IDF data-exchange to account for the resident behavioural aspect that define building system usage has been provided.

BIBLIOGRAPHY

[1] BIMERR, "Description of Action (DoA)," 2018.

[2] BIMERR, "Deliverable D3.6 - BIMERR system architecture final version," 2020.

[3] BIMERR, "Deliverable D3.3 - BIMERR evaluation methodology," 2019.

[4] ISO 16739, "Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries (ISO 16739:2013)," CEN, 2016.

[5] T. Hong, S. D'Oca, S. Taylor-Lange, W. Turner, Y. Chen and S. Corgnati, "An ontology to represent energy-related occupant behavior in buildings. Part II: Implementation of the DNAS framework using an XML schema," *Building and Environment,* vol. 94, no. 1, pp. 196-205, 2015.

[6] BIMERR, "Deliverable D5.8 - Building resident energy related behaviour profiling framework 2," 2021.

[7] BIMERR, "Deliverable D7.5 - Building Energy Modelling Module 1," 2020.

[8] BIMERR, "Deliverable D3.1 - Stakeholder requirements for the BIMERR system," 2019.

[9] BIMERR, "Deliverable D4.3 - BIMERR Ontology and Data Model 2," 2021.

[10] BIMERR, "Deliverable D5.2 - Prototype of enhanced BIM platform 2," 2021.

[11] BIMERR, "Deliverable D5.4 - Innovative Scan-to-BIM tools for Automated BIM generation 2," 2021.

[12] BIMERR, "Deliverable D8.3 - Integrated BIMERR ICT system 1," 2021.

[13] OptEEmAL, "Requirements and design of the simulation model input generator module," 2016.

[14] ISO 13790, "Energy performance of buildings — Calculation of energy use for space heating and cooling," 2008.

[15] EPBD, "Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings (recast)," 2010.

[16] D. Yan, J. Xia, W. Tang, F. Song, X. Zhang and Y. Jiang, "DeST - An integrated building simulation toolkit Part I: Fundamentals," *Building Simulation,* vol. 1, no. 2, pp. 95-110, 2008.

[17] D. Crawley, L. Lawrie, F. Winkelmann, W. Buhl, Y. J. Huang, C. Pedersen, R. Strand, R. Liesen, D. Fisher, M. Witte and J. Glazer, "EnergyPlus: creating a new-generation building energy simulation program," *Energy and Buildings,* vol. 33, no. 4, pp. 319-331, 2001.

[18] J. W. Hand, "The ESP-r cookbook," 2011.

[19] S. A. Klein, W. A. Beckman and J. A. Duffie, "TRNSYS - A Transient Simulation Program," *ASHRAE Transactions,* vol. 82, no. 1, pp. 623-633, 1976.

[20] P. Fritzson and C. Engelson, "Modelica - A unified object-oriented language for system modeling and simulation," *ECOOP'98—Object-Oriented Programming,* pp. 67-90, 1998.

[21] A. Andriamamonjy, D. Saelens and R. Klein, "An automated IFC-based workflow for building energy performance simulation with Modelica," *Automation in Construction,* vol. 91, pp. 166-181, 2018.

[22] G. I. Giannakis, K. I. Katsigarakis, G. N. Lilis and D. V. Rovas, "A Workflow for Automated Building Energy Performance Model GenerationUsing BIM Data," in *16th IBPSA Building Simulation Conference*, 2019.

[23] G. I. Giannakis, G. N. Lilis, M. A. Garcia, G. D. Kontes, C. Valmaseda and D. V. Rovas, "A methodology to automatically generate geometry inputs for energy performance simulation from IFC BIM models," in *14th IBPSA Building Simulation Conference*, 2015.

[24] J. Kim, W. Jeong, M. Clayton, J. Haberl and W. Yan, "Developing a physical BIM library for building thermal energy simulation," *Automation in Construction,* vol. 50, pp. 16-28, 2015.

[25] W. Jeong, J. B. Kim, M. J. Clayton, J. S. Haberl and W. Yan, "Translating building information modeling to building energy modeling using model view," *The Scientific World Journal,* vol. 21, 2014.

[26] EBC Annex 66, "Final Report-Definition and Simulation of Occupant Behavior in Buildings," 2018.

[27] G. Lilis, G. Giannakis and D. Rovas, "Automatic generation of second-level space boundary topology from IFC geometry inputs," *Automation in Construction,* vol. 76, pp. 108-124, 2016.

[28] J. O'Donnell, R. See, C. Rose, T. Maile, V. Bazjanac and P. Haves, "SimModel: A Domain Data Model for Whole Building Energy Simulation," in *Building Simulation 2011: 12th IBPSA Conference*, Sydney, 2011.

[29] BIMERR, "Deliverable D7.10 - Integrated BIMERR Renovation Decision Support System 2," 2021.