Preprints.org

Article

The Urban Building Energy Retrofitting Tool: An Open-Source Framework to Help Foster Building Retrofitting Using a Life Cycle Costing Perspective. First Results for Montréal

Oriol Gavaldà Torrellas [*] , Pilar Monsalvete , Saeed Ranjbar , Ursula Eicker

Article

The Urban Building Energy Retrofitting Tool: An Open-Source Framework to Help Foster Building Retrofitting Using a Life Cycle Costing Perspective—First Results for Montréal

Oriol Gavaldà-Torrellas [1,*,†], **Pilar Monsalvete** [2,†], **Saeed Ranjbar** [1] and **Ursula Eicker** [1]

[1]   Next-Generation Cities Institute, Concordia University, 2155 Rue Guy, H3H 2L9 Montréal, QC, Canada
[2]   Universidad de Cádiz
*   Correspondence: oriol.gavalda@concordia.ca
†   Current address: 2155 Rue Guy, ER-1431.35 H3H 2L9 Montréal, QC, Canada

Abstract: Highlights of the article: An open-source Urban Building Energy Model developed for the easy integration of research in the urban sustainability fields has been developed by the Next Generation Cities Institute at Concordia University. The paper proves its facility of use applying the framework to the development of a fast analysis tool for retrofitting buildings, and applies it to the island of Montréal, showing promising results. Building decarbonization is a major challenge for cities. Deciding which buildings to retrofit buildings, and when and how, is difficult, given the complex interaction between energy costs and investment requirements. Several tools have been developed in recent years to help public and private stakeholders with these decisions, but none cover aspects the authors think are fundamental. For this reason, an urban buildings retrofit tool was developed and is presented in this article. This new tool is based on a bottom-up approach, with all buildings simulated individually, considering aspects such as shading and adjacencies. As a second step, three scenarios with different levels of ambition were implemented in the tool, and the energy demand and emissions resulting from these scenarios were calculated. As a third step, the retrofitting scenarios' initial investment and operational costs were implemented using a detailed Life Cycle Costing (LCC) approach. A robust and scalable structure was developed and applied to calculate the LCC of various retrofitting scenarios in Montréal, which will be described in detail.

Keywords: urban building energy models; retrofitting; life cycle costs

---

1. Introduction

One of the most demanding segments of urban-scale decarbonization is building retrofits. The building sector accounts for more than 35% of global energy consumption and almost 38% of CO_2 emissions if we consider embodied energy [1].

The decision of what we do to existing buildings can significantly impact future CO_2 emissions. In the scientific community, there are significant doubts that demolishing and rebuilding net zero is a better option than retrofitting, because of the reduced CO_2 embodied emissions of the latter option [2–4], but a strong effort must be made to ensure that it can be attractive to all stakeholders.

According to [5], even after a 90 lifetime, retrofitting seems a better option than demolishing. In most cases, retrofitting seems to be the best option from a multi-factor social, environmental, and economic perspective. According to recent references, the stock of existing buildings will account for 75% of GreenHouse Gases (GHG) emissions, whilst new buildings will account for 25% (2021 World Green Building Council. WorldGBC Net Zero Carbon Buildings Commitment). According to recent references, the stock of existing buildings will account for at least 75% of GHG emissions while new buildings will account for 25% [6] by 2050.

In Canada, the retrofitting of low-rise residential buildings represents less than 1%, and, for commercial buildings, 1.4% of floor area per year [7]. However, although the goal seems clear for most policymakers in different countries, massive and large-scale retrofitting is not happening. Very interesting examples of successful case studies have been reported worldwide when life cycle costs are taken into account [8], but large-scale implementation is not successful for several reasons. The most significant reasons [9] are knowledge gaps, expertise gaps, lack of political commitment, inadequate policies, the speed to scale up, the huge necessary investment capital, and long returns on investment.

Some of those, especially the ones related to policymaking and the knowledge gap, linked to the lack of detailed life cycle information on the impact of these measures [10], could be overcome using tools such as Urban Building Energy Modeling (UBEM). The demand for low-carbon and energy-efficient urban development has made UBEM a crucial technique for evaluating the energy efficiency of buildings at the city scale. Reinhart and Cerezo Davila [11] defined UBEM as a bottom-up, meta-scale, physics-based engineering model that combines mass flow and heat models to forecast indoor–outdoor environmental conditions and energy consumption for groups of buildings. Considering interdependencies between buildings and their surroundings, such as the impacts of height, density, and radiation exchange on energy consumption, this goes beyond conventional Building Energy Modeling (BEM), which concentrates on individual structures [12]. By offering insights into energy consumption patterns at the daily, seasonal, and annual levels, UBEM helps urban planners and decision-makers assess energy-efficient designs, retrofit options, and renewable energy potential [13,14]. UBEM is a vital instrument for sustainable urban energy planning because of its applications in grid integration, urban planning, and carbon reduction strategies.

Therefore, an Urban Building Energy Retrofit Tool based on UBEMs could be a step forward thanks to its scientific, modular, and scalable approach and ease of application to any city, given the reduced datasets necessary to implement it in a first rough approach.

2. Literature Review on UBEM and Retrofit Tools

2.1. Existing Urban Modeling Tools

Many tools have been developed in recent years to model energy flows in cities, trying to simplify the use of and access to urban energy models but only covering partially the complexity of the energy system [15]. So-called UBEM aims to provide understanding of all aspects of a city's energy flows, from building energy "consumption" to on-site renewable "production", including traffic and waste management. Within this broad field, many have focused on building stocks, which is the aspect the authors will focus on in this article.

UBEM analyses building demands in their urban context, and it is important to consider their interactions within urban areas with other buildings and other urban area objects. In their review paper on urban models and tools, Allegrini et al. [16] pointed out the importance of building-to-building thermal and solar interactions, i.e., the importance of adjacencies between buildings and the shading of different buildings caused by neighboring surfaces, Urban Heat Island (UHI) effects, and urban vegetation, and their impacts on the results of urban energy system simulations.

This interaction between a building and other objects in an urban area, such as another building that causes shading of views from facades to the sky and sun, was considered for the first time in 2005 in the Sustainable Urban Neighbourhood modeling tool, `SUNtool` [17]. A thermal model of a building, coupled with Heating, Ventilation, and Air Conditioning (HVAC) was integrated into the tool, also including an urban-sensitive radiation model, the Simplified Radiosity Algorithm (SRA) [18].

Heavily inspired by `SUNtool`, but further refined with more rigorous thermal and behavioral models was the comprehensive software `CitySim`. Citysim was a step forward with respect to previous UBEM tools, and it used a Reduced Order Models (ROM) developed at EPFL in Switzerland. The tool also allowed for calculations with a simplified dynamic simulation tool and integrations with other tools, such as the transportation tool MATSIM [19].

Another tool developed later was Simstadt. SimStadt is an urban workflow engine written in Java and developed by the University of Applied Sciences Stuttgart [20] in 2013. SimStadt can use both simplified geometry (extrusion of footprint) and, when available, it can use City Geography Markup Language (CityGML) format for detailed geometry (Level of Detail (LoD) 2) when considering roof shapes. A study performed by [21] modeled about 1000 buildings in Rotterdam, Netherlands, using the geometry format of CityGML. SimStadt benefits from the direct integration of , which contains thermophysical building attributes in a machine-readable format as an extension of CityGML [22]. Like other UBEM tools, SimStadt requires several preprocessing steps of input data to be suitable for energy analysis and has a monthly output resolution. Finally, it does not account for transportation and mobility calculations and their connection with the UBEM. However, it provides life-cycle analysis using embodied carbon for building construction in a seamless tool [23].

At the same time, the Urban Modeling Interface (UMI) was developed at MIT in 2013, using EnergyPlus as the energy simulation engine. The input geometry used in the UMI is provided by Rhinoceros software [24].

CityBES was also developed running EnergyPlus as the simulation engine, but in this case, using the City Geography Markup Language (CityGML) format for 3D urban building modeling in LoD1 and LoD2 (considering roof shapes). It can also be linked to EnergyADE to use thermophysical building attributes [25].

Two other UBEM tools that use Resistance Capacitance (RC) models as a base engine are OpenIDEAS (an open-source platform) and CEA [26]. The former uses an interesting approach to structure the input data sources and workflows, to allow good input preprocessing, output post-processing, and workflows to happen outside the main structure of the tool. CEA is one tool that integrates additional aspects, including several input sources, incorporating cost–benefit analysis, transportation use cases, and greenhouse gas emissions. Shadow analysis is also incorporated.

Urban Renewable Building and Neighborhood Optimization (URBANOPT) is one of the latest tools developed by the National Renewable Energy Laboratory (NREL). This tool uses EnergyPlus and is open-source [27,28]. It also contains lighting simulation but does not include transportation/mobility, life cycle, or cost–benefit analysis.

The Tool for Energy Analysis and Simulation for Efficient Retrofit (TEASER), developed at Aachen University (RWTH), is based on RC models, and it considers several carbon emission accounting strategies [27] .

In recent years, new developments have been made. An example of a tool that incorporates a similar approach to the one that the authors have undertaken in recent years is AutoBPS, a tool for urban building energy modeling to support energy efficiency improvement at city-scale [29–31]. Two other interesting recent developments in this field are CityBEM [32], based on self-developed heat transfer equation methods and UECC (a novel resistance–capacitance model for evaluating urban building energy loads considering construction boundary heterogeneity; a novel and efficient method for calculating beam shadows on exterior surfaces of buildings in dense urban contexts) [33,34], based on reduced-order models (R.O.M). In parallel, an interesting approach to infer occupancy for UBEMs using mobility-related data has been introduced by the tool Dest-urban (Development of a Platform of Urban Building Energy Modeling Based on DeST) [35,36]. A chronology to tools is shown in Figure 1.
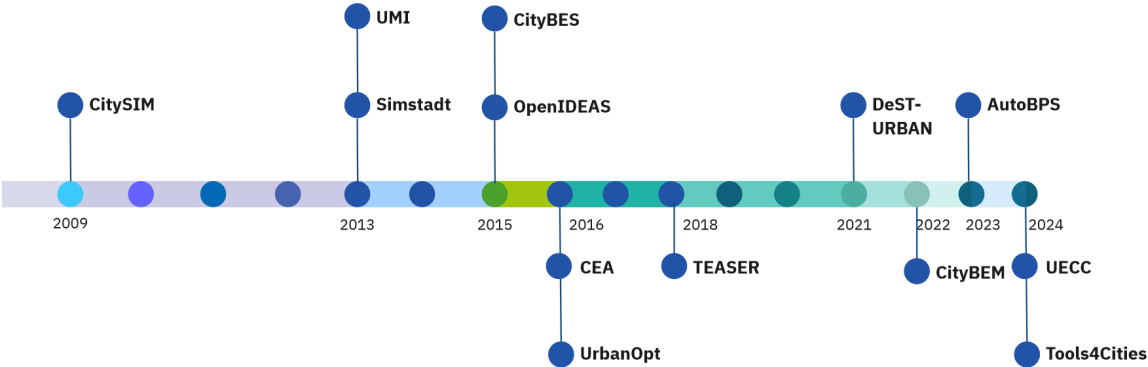
Figure 1. UBEM development timeline.

A summary of the different UBEM tools and their main aspects has been elaborated and is shown in Tables 1–**??**.

Table 1. UBEM tools and fields of work.

Tools	Buildings Operational	Buildings Life Cycle	Transport	Microclimate	Waste	Other Aspects
SUNTOOL	Yes	No	No	Yes	Yes	No
CITYSIM	Yes	No	Yes	No	No	No
SIMStadt	Yes	Yes	No	No	No	No
UMI	Yes	Yes	Yes	Yes	Yes	Food production, daylighting
CityBES	Yes	No	Yes	No	No	No
OpenIDEAS	Yes	No	No	No	No	No
CEA	Yes	Yes	Yes	Yes	Yes	No
UrbanOPT	Yes	No	No	No	No	Daylighting
TEASER	Yes	No	No	No	No	No
Dest-Urban	Yes	No	No	No	No	No
AutoBPS	Yes	No	No	No	No	No
UECC	Yes	No	No	No	No	No
CityBEM	Yes	No	No	No	No	No

Table 2. UBEM tools and input formats (based on all the mentioned references).

Tool	Geospatial Input Formats Accepted	Constructive Inputs	Usage Inputs Defined	Shading Analysis	Microclimate Possibilities	Other Aspects
SUNTOOL	Bespoke interface	To be defined by users	Static, deterministic and probabilistic	Yes, SRA	Yes	No
CITYSIM	CityGML, EnergyADE	To be defined by user, based on archetypes	Static, deterministic	Yes, SRA	No	No
SIMStadt	CityGML, EnergyADE	To be defined by user, based on archetypes	Static, deterministic	Yes, SRA	No	No

Table 2. *Cont.*

Tool	Geospatial Input Formats Accepted	Constructive Inputs	Usage Inputs Defined	Shading Analysis	Microclimate Possibilities	Other Aspects
UMI	Rhino3D	To be defined by user, based on archetypes	Static, deterministic	Yes, based on Eplus	Yes (UWG)	Food production, daylighting
CityBES	CityGML, EnergyADE	To be defined by user, based on individual buildings	Static, deterministic	Yes, based on Eplus	No	No
OpenIDEAS	CityGML, EnergyADE	To be defined by user, based on archetypes	Static, deterministic and probabilistic	Not well explained and detailed (missing?)	No	No
CEA	CityGML, osm	To be defined by user, based on archetypes	Static, deterministic and probabilistic	Yes	Yes, EnviMET can be coupled to the models	No
UrbanOPT	CityGML, GeoJSON, osm, IDF	To be defined by user, based on archetypes	Static, deterministic	Yes, based on Eplus	No	Daylighting
TEASER (part of IDEAS)	CityGML, EnergyADE	To be defined by user, based on archetypes	Static, deterministic and probabilistic	Not well explained and detailed (missing?)	No	No
DeST-URBAN	Own processing	To be defined by user, based on archetypes	Very detailed. Static, deterministic and probabilistic	Based on archetypes	No	No
AutoBPS	CityGML, EnergyADE	To be defined by user, based on archetypes	Static, deterministic and probabilistic	Not well explained and detailed (missing?)	No	No
UECCC	Not described (seems obj)	To be defined by user, based on archetypes	Static	Detailed shading, based on previous work	Not described	No
cityBEM	obj	To be defined by user, based on archetypes	Static	Yes	No	

Table 3. USEM tools and software engine characteristics (based on all the mentioned references).

Tool	Engine for Building Simulation	Engine for Energy Systems Calculations	Engine for Mobility Calculations	Programming Language	Output Processing	Open-Source Architecture	Frequency Data
SUNTOOL	R.O.M (grey-box model)	Yes	No	Java	No	No	Hourly
CITYSIM	R.O.M (RC model)	Own engine	MATSIM-c	C++ and java	Yes	Yes	Hourly
SIMStadt	Simplified energy balance methodology	Own engine	No	Java	Yes	Yes	Monthly

Table 3. *Cont.*

Tool	Engine for Building Simulation	Engine for Energy Systems Calculations	Engine for Mobility Calculations	Programming Language	Output Processing	Open-Source Architecture	Frequency Data
UMI	Energyplus	Energyplus	Own engine (python), and walkability based	Rhino Grasshopper (python)	Yes	Partly dependent on Rhino.	Hourly and sub-hourly
CityBES	Energyplus	Energyplus	No		Yes	Yes	Hourly and sub-hourly
OpenIDEAS	R.O.M (RC model)	Modelica	No	Python	Yes	Yes	Hourly
CEA	R.O.M (RC model)		No	Python	Yes	Yes	Hourly
UrbanOPT	Energyplus	Energyplus	No	Ruby, C++, Python	Yes	Yes	Hourly and sub-hourly
TEASER	R.O.M (RC model)	Modelica	No	Python	Yes	Yes	Hourly
Dest-URBAN	Energyplus	Enerygplus	No	Python	Yes	Yes	Hourly
AutoBPS	Energyplus	Energyplus	No	Python	Yes	Yes	Hourly
UECC	R.O.M (RC model)	Own	No	Own tool	Yes	Yes	Hourly
CityBEM	Own heat transfer model	Not considered	No	CSharp	Yes	Yes	Hourly

Table 4. USEM tools and scenario creation (based on all the mentioned references).

Tools	Incorporation of Cost/Benefit Analysis	Incorporation of Scenarios	UI Tool	Level of Scaling Up of the UBEM Tool
SUNTOOL	No	No	Yes	Low scaling-up
CITYSIM	No	No	Yes	Low scaling-up
SIMStadt	No	Yes	Yes	High scaling-up
UMI	No	Yes	Yes	High scaling-up
CityBES	Yes		Yes	Low scaling-up
OpenIDEAS	No		No	Low scaling-up
CEA	Yes	Yes	Yes	High scaling-up
UrbanOPT	No	Yes	Yes	High scaling-up
TEASER	No		Yes	Low scaling-up

2.2. Gaps to Be Overcome

The different UBEMs described in the previous sections have established seeds for deploying relevant future urban system energy tools.

However, several research questions arise from the literature review and the tables mentioned. The capacity of these tools to answer real cost–benefit questions from the building decarbonization sector is often in doubt for several reasons. When dealing with such complex problems, the usual question is whether the UBEMs were designed with a goal in mind and whether they will be useful for more than this goal.

The capacity, therefore, of the aforementioned UBEMs for adaptability and modularity is small. Although they try to be multi-faceted tools, the design of a tool, if not considered modular from the beginning, tends to be adapted to the use case for which it was first developed. Each UBEM was

developed with a back-end calculation methodology in mind, which is applicable in the field for which this methodology is useful.

Another important aspect is the lack of scalability to a large number of buildings and different locations. For example, most of the aforementioned tools have been applied to one or a few (under 5) use cases and, in most cases, linked to a small set of buildings. The use, in some cases, of simplified reduced-order models (Suntool, CitySIM, OpenIDEAS, CEA, TEASER), and the quality of the results with the use of very simplified models, makes it difficult to extend to larger scales. Conversely, using detailed dynamic simulation tools (UMI, CityBES, UrbanOPT) makes it hard to develop large-scale scenarios.

These tools' lack of capacity to deal with several types of input datasets (as seen in Table 2) makes it necessary to clean and adapt the input data sources every time. This is a highly cumbersome process that sometimes takes months and limits the scaling up of the methodology. The typically incoherent input datasets from cities also make it necessary to compare several sources of information and choose the cleanest, which, in the cases of most current UBEMs, is a process that has to happen before using the tools.

The fourth is the lack of data input automation. Many authors [15,16,37–39] have delved deeply into the relevance of data input automation, which reduces the chance of input errors and also human costs, as the data processing is, in most cases, one of the highest time consumers [40]. Lack of open-source developments is also an issue, at least for some aspects of the different UBEMs. UBEMs deal with very different sectors of GHG emitters (buildings, transportation, water), and breaking down silos and integrating attractive models from academia in all those fields should be the goal. The development of UBEMs using proprietary software can partially block researchers from accessing these tools.

As a last point, clear communication of results has long been an issue in using urban simulation tools [16]. In the urban simulation tool field, GIS thematic mapping becomes a very intuitive tool for final users to access and understand results [41–43].

2.3. Reasons for the Development of a New Model

The research question that arises from the literature review and the gaps detected is whether the current UBEM tools are designed to address the challenges required by different use cases. Is the structure behind the development of each UBEM conditioned by the use case for which each UBEM was developed? After reviewing the previous chapter, we found a clear link between the use case and the structure of the tools.

The current tools have developed ad hoc solutions to enter city-based data, populate them with structured information about building archetypes, and directly produce the necessary building simulation input. Some of them have done so using open-source structures, but others have focused on their own engines and development. Moreover, very few work in different sectors (such as buildings and transportation), and even fewer (only UMI) link the results of those different sectors.

The main problems lay, therefore, in the flexibility of these tools to capture different types of data sources and to use different simulation tools. Using different simulation tools is crucial for working with different sectors and working at different levels of detail in each sector. The newly developed model is, thus, based on creating a central data model with a clear definition of all the processes in cities that could be used for decarbonization. This central data model is where different sources of information would be stored and structured, and the structure could produce the necessary input files for different typologies of software, depending on the use case. It would also be a framework that could allow different processes to store information and chain different software, using the output of one process as an input for the following processes.

Therefore, the answer to all these gaps is creating an open-source, flexible, and structured framework that allows multiple types of input files of city characteristics and tools to be used and integrated.

This framework should allow for the development of different use cases, all related to improving the sustainability of cities. The non-adherence to any concrete software, but the decision to create a framework to integrate them all is essential for this purpose. Some tools will be preferable to others when analyzing several use cases, and the platform will deal with them.

Thus, the research question amounts to understanding the essential structure of an urban energy modeling tool with limited data that can help foster retrofitting the building stock in cities.

As an answer to this question, the framework described in the present paper, the TOOLS4Cities hub, has been developed in the last three years, with the application of this framework to the concrete case of an urban scale retrofit tool.

3. Materials and Methods

The previously defined TOOLS4Cities UBEM framework was the basis for developing a tool to analyze the potential of deploying large-scale retrofitting strategies in cities. The current paper will explain the structure of the retrofit tool and some aspects of the software architecture. The goal is to show the framework's basic structure and prove its modularity and scalability with a use case. The details of the implemented software structure are outside the scope of this paper, and interested readers can refer to Appendix A to familiarize themselves with details of the software architecture. The developed code is open-source and available for cloning in git [44].

3.1. General Overview

A general overview of the retrofitting workflow is shown in Figure 2. The workflow is part of the TOOLS4Cities urban simulation platform, developed at the Next Generation Cities Institute of Concordia University, aiming to simulate urban areas holistically. The main part of this simulation platform is a set of pipelines, data models, datasets, and catalogs called the TOOLS4Cities Hub. The building retrofit tool comprises building modeling, energy system modeling, scenario selection, techno-economic assessment of scenarios, and visualization modules using different parts of the TOOLS4Cities Hub. These modules will be explained in detail in the following sections.
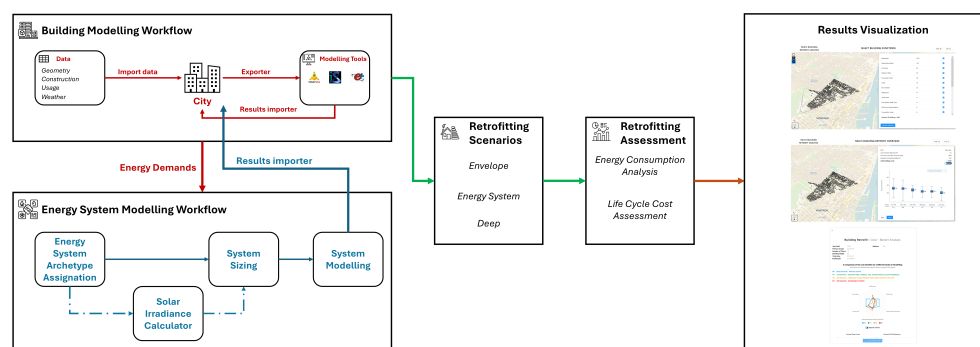

Figure 2. Retrofitting workflow overview.

The retrofit workflow's essential objective is to analyze the energetic, environmental, and economic impacts of retrofitting any building in a city. The idea is to pick any building in a city and obtain the investment costs, operational costs, and environmental emissions of leaving the building as-is or implementing some pre-defined retrofitting strategies. In the first approximation (or Minimum Value Product (MVP)) of the retrofit workflow, three basic retrofitting strategies were implemented (skin retrofit, energy systems retrofit, and deep retrofit) for all the buildings of the city. However, the tool's structure allows for more detailed large-scale retrofitting strategies and heuristics, to apply different strategies depending on different typologies of buildings. As an example, the oldest buildings might require a complete skin retrofit, but maybe the newest will only require changing windows or retrofitting the energy systems.

3.2. Building Energy Modeling Workflow

For good scalability, a comprehensive software structure had to be developed to allow for the evolution of the datasets and the adaptation to different tools, while maintaining a robust ontology. As shown in Figure 2, the UBEM structure of TOOLS4Cities consists of importers, exporters, and

a representation of the 'City' with as many objects, such as buildings, green spaces, roads, etc., as come from the input file the user provides. The creation of this city replica is made possible through object-oriented programming, where a 'City' class includes different attributes, some of which are objects from other classes that represent different city objects. Therefore, from this point onward, in this paper, when 'City' is referred to, it is an object from the 'City' class in the tool's backend, which can have zero to many buildings and other real objects in any urban area.

The objective in developing the UBEM workflow was to provide a platform where the input file for running different simulation tools like EnergyPlus or INSEL could be exported, and these tools could be run externally. The handling of the data required is made possible through different data models and catalogs [45]. A data model is a formalized framework that defines how data are structured and represented within a system. It is distinct from the data and does not dictate where they are physically stored. Instead, it serves as an abstract blueprint that helps us understand the relationships and interactions among all the data components within the system.

At the core of the platform are the data models. Addressing the gaps mentioned above in existing UBEM models highlights the critical need for a well-organized data structure. Such a structure is essential to facilitate clear and effective interactions among the various sectors involved in urban decarbonization use cases. Furthermore, it will enable UBEM tools to function as versatile, multi-purpose instruments, allowing for adaptability and customization of the final tool to meet evolving requirements. The following sections explain what data are needed and how they are used in the UBEM workflow.

3.2.1. Building Geometry

The first step in modeling buildings is creating a 3D city model. In the TOOLS4Cities Hub, a 3D model can be generated from different formats, including GeoJSON and CityGML. In the case of buildings, the input file must have information about the year of construction, building function, and height. First, an instance from the 'City' class is created in this process. Then, it is populated with the city object information from the input file, using a specifically designed importer to enrich the city with data from the geometry file.

Once the extruded model of the city objects has been created, the internal zones of buildings are defined. Currently, each building has a single internal zone, but different modes of multi-zoning are being developed.

3.2.2. Construction

After creating the 'City', the buildings need to be enriched with physics or construction data to create the thermal zones inside the building. The whole process of urban modeling in TOOLS4Cities is archetype-based, meaning that the buildings are enriched with construction data purely based on their construction archetype. Therefore, before starting the UBEM process, the sets of archetypes need to be created. The standards followed by the Next-Generation Institute in distinguishing archetypes are based on their function, period of construction, and climate zone. Therefore, these standards must be followed in new case studies where archetypes still need to be created in the TOOLS4Cities Hub.

3.2.3. Usage

The usage archetypes contain data on a building's internal gains and profiles. These archetypes are assigned to a building solely based on the building's function. The profiles refer to the heating and cooling setpoint schedules, HVAC availability, occupancy, lighting, appliance usage, and domestic hot water consumption. For each category, three sets of schedules are defined and used for weekdays, Saturdays, and Sundays/holidays. The schedules contain hourly fractions of the density values for the mentioned categories.

3.2.4. Weather

Climatic data are an essential part of building modeling. Hourly outdoor temperature, direct normal irradiance, diffuse irradiance, etc., values are read from a weather file that could come from any source. If the user provides no weather file, that location's '.epw' file will be downloaded automatically and used for the simulations.

3.2.5. Modeling Tools

Once the 'City' buildings are enriched with the required data for modeling, a factory of different exporters can export the input file for different simulation tools. The exporters capture the necessary data from the central data model, which describes the city's parameters, and convert them into input data files for different software. At the moment, version 2.0 of the framework can export data models to the following simulation tools:

- Simplified Radiosity Algorithm (SRA), a tool designed to calculate the plane of array irradiance on all surfaces, allows for a detailed radiation calculation to deploy solar photovoltaic analysis.
- INSEL [46], a tool which is capable of calculating the monthly and yearly energy demands for the buildings of a city using a static and dynamic methodology
- EnergyPlus, one of the best known and most extensive tools for calculating the energy demand and consumption of buildings
- TRNSYS, another one of the reference tools for building and energy system simulation.

The process includes running methodologies, which run the different tools and then store the results back in the central data model, as the results paragraph will define. Although some of the tools have the same objectives (TRNSYS, Insel dynamic, and EnergyPlus), the inclusion of the different exporters is intended to facilitate the integration of different research teams with the methodology, as well as to compare the capacities of each software and be able to pick any of them depending on the use case. As an example, the capacity of TRNSYS or INSEL to integrate external sources of radiation seems interesting as an option for integrating the SRA methodology into a simulation engine. EnergyPlus, on the other hand, has an internal radiation methodology.

EnergyPlus Workflow

Given that the retrofit costing methodology only involves using Energyplus, we have explained in more detail the methodology used for Energyplus. One of the aforementioned gaps is the scalability to a large number of buildings, and one of the main reasons for this is the long computing times of running white box models. One of the software developments to speed up this process has been the Energyplus workflow. The EnergyPlus workflow exports the central data model with all details of the buildings to an IDF file. Buildings are modeled as one-zone models in the current version. To speed up the process, buildings are created in the IDF as zones of the building, including the shadings and adjacencies between the different zones. The eppy library was initially used to create IDF files. However, the software characteristics of the library (repeating the different schedules per zone and having to access extended dictionaries) meant that when simulating more than 500 buildings, the IDF creation was significantly slowing down the process. Therefore, a new library was developed to reduce the export time. It will have to be updated for different versions of Energyplus, but it allows for large-size IDF exporting, which is essential for urban simulations. Although the main reason for the use of simplified models is the time used in the simulation of white box models, the current improved process can simulate up to 3000 buildings in less than 2 h using a $1\times$ Intel® Xeon® E-2400 processor, with up to 8 cores at 95 W.

SRA

The simplified radiosity algorithm used by the tool uses the methodology developed by [17], which applies radiosity to predict external irradiance (shortwave and longwave) and internal illuminance in an accurate and very efficient way, in a single computational module. Although the internal

radiation calculation engine is used for the Energyplus simulation, the SRA is used to calculate the radiation on surfaces to implement solar PV calculations. The SRA importer makes use of the imported geometry and the physical characteristics of the materials of the external layers of the walls, together with weather data, to calculate the shortwave radiation exchange between external walls and roofs.

3.2.6. Results

After running the simulations, the results importer is used to enrich the buildings with the results of the simulations. This step helps us to use the 'City' structure as it is for further analysis but, this time, buildings have energy demands.

3.3. Energy System Modeling

For each location, research is performed to identify the dominant energy systems used to supply HVAC and DHW demands. These systems comprise energy generation, distribution, storage, and heat emission components. All these systems together form an energy system archetype or template. This structure is the basis of the energy system data model in the TOOLS4Cities Hub.

The created data model, Figure 3, functions as a thorough and well-structured framework, intended to record and handle the intricacies of energy system setups methodically. The 'EnergySystemCatalogue', a conceptual entity that functions as a flexible library with multiple subclasses, is at the center of the model. These subclasses, divided into areas including distribution, generation, storage, and emissions, are essential for providing specifics about particular components. The versatility of the model is demonstrated by its capacity to represent components in two different ways: real-world components available on the market or template structures with predefined attributes and no assigned size. This two-pronged strategy produces a flexible framework that can handle various situations.
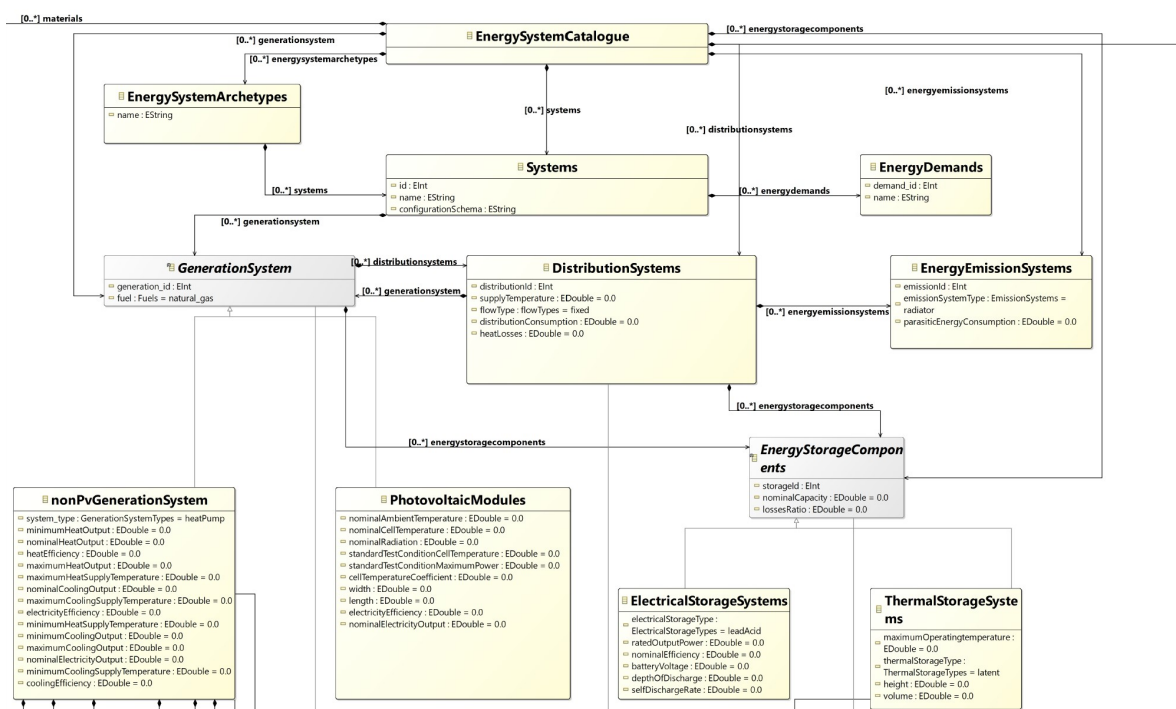

Figure 3. A snippet from the UML diagram of the energy system data model. In the image, * means a number n of potential cases.

Once the energy demands have been calculated, an energy system template is assigned to the building based on its function and vintage. The associated data of the template components, such as the efficiency of the generation components, are retrieved from different '.xml' files that contain energy system data. An importer enriches buildings with data that can be easily used to calculate energy consumption in any area.

3.4. Retrofitting Scenarios

Although the possibilities of retrofitting are infinite, three retrofit scenarios for the first retrofit workflow were established as a test case to validate the full methodology. These scenarios were the following:

- Scenario 0, Business as usual: current status of the building, with its skin, but replacing the energy systems once their lifetime is complete (with the same performance as the original).
- Scenario 1, Skin retrofit: the building's envelope is updated to current legislation.
- Scenario 2, HVAC systems optimization + PV: an update of HVAC systems to the best available technologies is carried out, coupled with using the available roof space for Solar Photovoltaicss in cases where the shaded area is less than 30% of the total area.
- Scenario 3, Deep retrofit: a scenario including both previous scenarios.

3.5. Costing Workflow

In the case of the Life Cycle Costing (LCC) process, we dealt with this in a separate classes structure. The different nature of economic hypotheses, which are much more volatile than energy calculations, has pushed the team to develop a parallel structure coupled with the hub structure. This parallel structure will allow for further analysis, from the perspectives of economic scientists, such as implementing dynamic cash flow modeling with sensitivity and risk analysis, a topic that has been detected as essential [47], or monetizing the externalities. The methodological approach for the economic analysis of the retrofit scenarios comprises the general LCC framework approach and the evaluation of externalities included in the general cost modeling methodology. LCC analysis, also known as total cost accounting, compares the full costs during the useful life, while defining alternative investment options in buildings and operating activities. For example, a baseline case can be compared with a potentially attractive one using the LCC of a retrofit scenario. LCC can showcase the different cost allocations of both scenarios, given that one of them already exists and does not imply extra investment costs. The new proposed scenario implies an extra investment but, on the other hand, minimizes yearly costs and, potentially, other externalities. LCC analysis design and practice is standardized in ISO 15686-5 [48], establishing the necessary criteria for analyzing life cycle costing in the built environment. The global formulation is shown in Equation (1).

$$
\begin{aligned}
LCC = CAPEX &- Subsidies + \\
\sum_{i=nyears} &\frac{\sum_{k=chaptinvest} CAPEX_{repositionk}(1+cpi)^i}{(1+d)^i} + \\
\sum_{i=nyears} &\frac{\sum_{j=nfuels} Operationalcostsj(1+epi)^i}{(1+d)^i} + \\
\sum_{i=nyears} &\frac{\sum_{l=nconcepts} Maintenancecostsl(1+cpi)^i}{(1+d)^i} + \\
\sum_{i=nyears} &\frac{\sum_{m=nincomes} Operationalincomesm(1+epi)^i}{(1+d)^i} + \\
\sum_{i=nyears} &\frac{\sum_{n=nextern} Externalitiescostsn(1+extpi)^i}{(1+d)^i} + \\
&\frac{Endoflifecosts(1+cpi)^i}{(1+d)^i}
\end{aligned}
\tag{1}
$$

where
$CAPEX$ = initial investment (product acquisition + installation and commissioning);
$CAPEX_{reposition}$ = investment cost for the reposition of the equipment after the end of life of the equipment;
$Operationalcosts$ = refers to the variable costs of operation;
$Maintenancecosts$ = refers to the variable costs of maintenance;
$Operationalincomes$ = refers to the variable incomes of operation, i.e., solar PV feed-in tariff;
$Externalitiescosts$ = refers to the variable costs of externalities, like CO_2 or other emissions;
cpi = consumer price index increase. Series of yearly values
epi = energy price index increase, which depends on the fuel type. series of yearly values

extpi = externalities price index increase, which depends on the future externalities market, such as CO_2 markets

d = discount rate for the retrofitting

CAPEX calculation: In our tool, we use a data catalog structure to calculate the CAPEX costs. In the case of CAPEX, UNIFORMATII was chosen as the format to classify building elements and related site work [49]. The level of detail used in the first validation of the tool was significantly low, as the solutions were looked at from a global perspective. Within this LoD1, and for the sake of the first tests, only the additional costs of the retrofitting were considered, with the average values per surface of the different elements and not distinguishing different solutions. The first structure of the data catalog for CAPEX costs is shown in Table 5.

Table 5. Breakdown of UNIFORMATII costs used in the platform.

Id	Concept
B	Shell
B10	Superstructure. Ground refurbishment
B20	Envelope
B2010	Opaque walls
B2020	Transparent walls
B30	Roofing
B3010	Roofing opaque
B3020	Roofing transparent
D	Services
D30	HVAC
D3010	Energy supply (PV systems)
D3020	Heat generating systems
D3030	Cooling generating systems
D3040	Distribution systems
D3060	Control and instrumentation
D3080	Other HVAC systems. AHU
D50	Electrical
D5020	Lighting and branch wiring
Z	Allowances
Z10	Design allowance
Z20	Overhead and profit

3.6. Visualization of Outputs

Significant effort has been made in the presentation and accessibility of the results for the general public. As mentioned in 2.2, this is one of the largest gaps we detected in other tools.

A web page showing all the buildings in the city is used to facilitate access to the city-simulated data and let the user interact with the data. The results are shown using two different services in the Citylayers test web page (https://nextgenerations-cities.encs.concordia.ca/citylayers/; accessed on 20 January 2025).

4. Results and Discussion

4.1. Case Study: Montreal

The first proof of concept for developing a retrofit calculation for an entire city was implemented on the island of Montréal. Over time, various geospatial, construction, usage, and system data were collected for Montreal, to enable the tool development team to create different archetypes. To validate the process, the authors focused on two residential areas and a set of institutional buildings, comparing the tool results with statistical and real operational data.

4.2. Selected Areas

To validate the surface and consumption for single-family houses, duplexes, and triplex, we focused on two large low-height residential areas in Montréal. The choice of the two sectors was to find homogeneous areas that could be compared with a concrete typology of building, for which we could have statistical data. Although, in the two chosen neighborhoods, there are some taller buildings and some commercial buildings, more than 95% of the built surface in both is occupied by single-family, duplex and triplex buildings. The first is in the Pointe-Saint-Charles neighborhood (between Wellington, Ash, Leber, and Congrégation streets), with a total of 313 buildings, most of which were built before 1946. The second subset of buildings is located in the Saint Leonard neighborhood (between Langelier Boulevard, Robert Boulevard, de Guyenne Street, and Louis Sicard Street), with a subset of 249 buildings from the 1960s and 1990s. Figure 4 shows the zones that were simulated. This gave rise to a total of 563 buildings.

Figure 4. Simulated buildings, color-coded by vintage.

Following the results of the first simulations, we analyzed in detail the origin of the differences between the simulated and actual consumption. The sources of error were analyzed from two points of view: (i) input data and (ii) simulation assumptions. In the first stage, the analysis focused on

discrepancies in the area calculated by our tool, based on the city's various geospatial data sources (CityGML, geojson, and land valuation data) and consumption per square meter.

The validation was carried out in two steps: the first was the consumption data analysis for specific typologies of buildings for which we had available metadata. We analyzed the data about two typologies of buildings: single-family homes/duplexes and public buildings.

The first level of validation checked whether the yearly results from the two use cases matched the real yearly consumption data. The sources of the data used as a comparison set were different.

For the single-family homes/plexes, the values to which the urban simulation was compared were a set of 19.111 buildings from the Energuide dataset [50], filtered only for Montréal and for single-family/duplex/triplex houses. This dataset gives yearly final energy consumption data for all the houses, and metadata for the year of construction of the buildings, and the typology of buildings is described in the metadata.

For tertiary sector buildings, we analyzed the list of 186 municipal buildings published by the City of Montreal, with metadata and their energy consumption [51]. For the process of validation, we focused on a subset of only 15 buildings.

4.2.1. Geometry Data

The geospatial data cleaning process was extremely time-consuming, especially because of the deficient quality of the input datasets and the necessary data cleaning. The dataset was created by combining several open-source datasets provided by NRCan and the Montreal city government (Montreal property assessment units [**?**], Montreal 3D buildings (LoD2 model with textures), LiDAR aerial dataset of Montreal [53], Building Footprints NRCAN [54], and Layer 2 cadastral data for disaggregated building use types [55].

However, even if the initial datasets were quite complete, the differences in quality and completeness forced us to follow specific necessary steps, such as (i) splitting the satellite-derived footprints based on information from the city's cadastral parcel data; (ii) correcting building heights using LiDAR data and improving the accuracy of the analysis; and, especially, (iii) incorporating an algorithm to detect and clean the overlap between the buildings. The final integrated result was a geojson database, with corrected building heights and an essential number of characteristics for each building (vintage; function, out of a list of more than 250 functions; number of households; number of stories). This geojson database was finally refined with a methodology to recognize neighboring walls in 2D and 3D. This process was used to generate a unique geojson file that contained all the necessary elements for the building simulated. The data were validated by comparing the number of households by postal code from the census of Canada and the number of households resulting from the footprints and the heights. The difference in all the neighborhoods, except for Nun's Island, was less than 5% of the total number of households. The quality of this set of data from the city was poor. The geometry factory from the TOOLS4Cities used the footprint and height data to generate 3D surfaces for all the buildings. Depending on the function and vintage of the building, a window-to-wall ratio was assigned to each of the buildings; the same process was applied to the division of buildings into stories, to understand the total surface of the buildings. An essential aspect of the process was that all the surfaces in contact with other surfaces were considered neighboring surfaces, and they were simulated as such for radiation and thermal purposes. One of the important aspects of the geometry factory was assigning a building archetype to each of the functions of the Montréal dataset of functions. The tax evaluation data describe more than 500 building functions [52], and they were converted into the set of archetypes developed in the hub. The archetypes used were based on the 17 archetypes developed by NREL and used by the BTAP tool [56] developed by Natural Resources Canada.

4.2.2. Construction Data

As explained in the general methodology, the tool uses a dictionary of archetypes that, given the vintage of each building, populates the constructive characteristics of each building. The first archetypes developed for Montréal were based on the developments of NRCAN for the BTAP tool,

which inherited a significant amount of work carried out by NREL. Those archetypes, however, have a limited amount of vintage categories regarding constructive solutions. For example, all the buildings before 1980 (which, in Montréal, account for more than 65 percent of the overall surface) are considered to have the same construction type. For that reason, several sources were consulted, and, based on an initial effort using BTAP datasets [56], we developed new constructive types based on references from academic studies [57,58] and analysis from consultancies [59]. The result is a compilation of different constructive types depending on whether the building is residential or non-residential and on the vintage of the building. The detailed constructive types used for the Montréal use case (and also developed for other climates) can be found in Tables A1 and A2, in Appendix B.

4.2.3. Usage Data

For Montreal, the National Energy Code for Buildings of Canada dataset, which uses fixed profiles for each building, was used to populate the usage data for all types of buildings and spaces, based on the previously defined archetypes per building. We used two steps, one for the gains and another for the profiles. These separate steps currently use a direct link to the repository of [56], where the NRCAN team has been updating the base data sources for the tool BTAP. As detailed in Appendix A, using public online data was a decision taken by the authors to reduce the maintenance costs of the Tools4Cities hub and allow for continuous updating.

Moreover, the factory structure already allows the integration of external stochastic generation profiles derived from [45], although it was decided not to use these for the retrofit use case.

4.2.4. Energy System Data

For the latter, the yearly balance in each sector was used to divide the buildings by gas and electricity consumption. Regarding the systems typologies, the NECB equivalence [60] from table Table 8.4.4.7.-A was used. These typologies show that a concrete HVAC system typology was suggested for every type of building. The systems, from System 1 to System 6, are shown in Table 6.

Table 6. Description of the different considered energy systems, based on the NECB.

Systems	System	Fan Control	Cooling System	Heating System
System 1	Unitary air conditioner with baseboard heater	CV	Air cooled DX with external condenser	Fuel-fired water boiler with baseboards
System 2	4-pipe fan-coil	CV	Water cooled, water chiller	Fuel-fired or electrical resistance water boiler
System 3	Single zone packaged rooftop unit with baseboard heaters	CV	Air-cooled DX	Fuel-fired or electrical resistance furnace, hot water or electrical resistance for baseboards
System 4	Single zone make-up air unit with baseboard heating	CV	Air-cooled DX	Fuel-fired or electrical resistance furnace, hot water or electrical resistance for baseboards
System 5	Two pipe fan-coil	CV	Water cooled, water chiller	None
System 6	Multi-zone built-up system with baseboard heater	VV	Water cooled, water chiller	Baseboards : electrical resistance or hydronic with fuel boiler

The main challenge in using the table above is that the energy source is not mentioned. Therefore, another data source was used to find the primary energy sources used in building consumption in

Quebec. Figures 5 and 6 show the share of different fuels in the heating system from the Survey of Household Energy Use (SHEU) conducted by Statistics Canada [61] and from the [62] for commercial buildings.
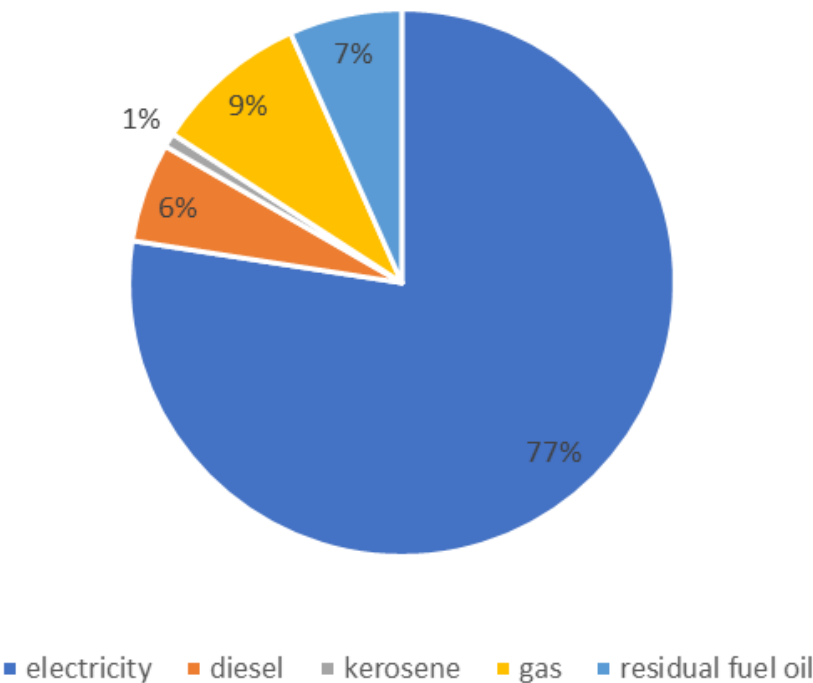
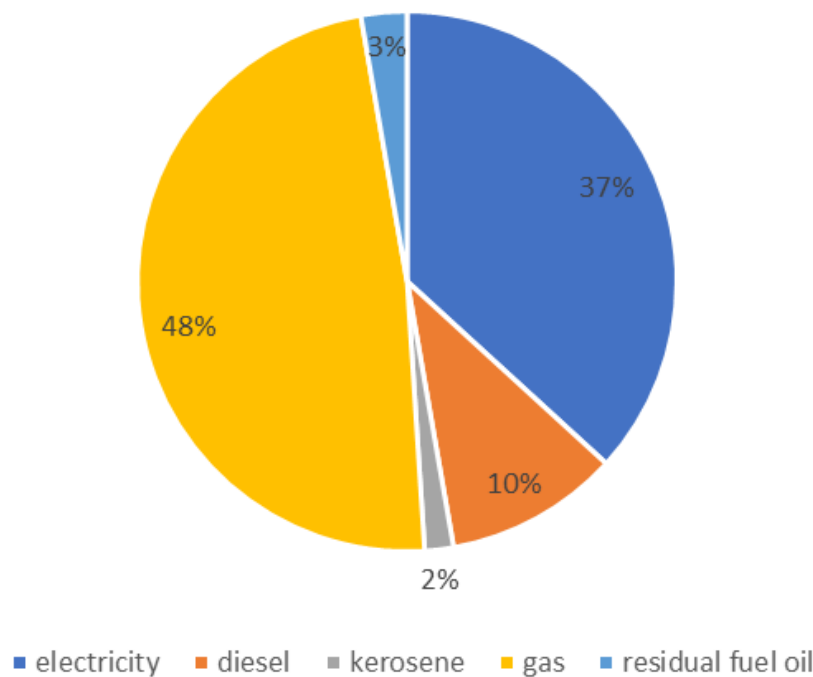**Figure 5.** Split between different data sources for residential buildings.

Figure 6. Split between different data sources for commercial buildings.

4.3. Geospatial and Energy Consumption Validation for Montreal

4.3.1. Single Family Houses/Duplex/Triplex

The results concerning the building area inferred from the geospatial inputs showed higher values than the actual values obtained through the property valuation data. The fact that the methodology captured the 3D exterior volume resulted in errors such as overestimating the surface area of second floors or incorporating partially covered outer volumes into the analysis. Figure 7 shows that the

average floor area calculated with the models was 60% higher than the values extracted from the property data. The lack of good data was at the root of this error. Some novel methodologies are being applied to detect cleaner methodologies via AI and ML and these will be incorporated into future platform developments [63].
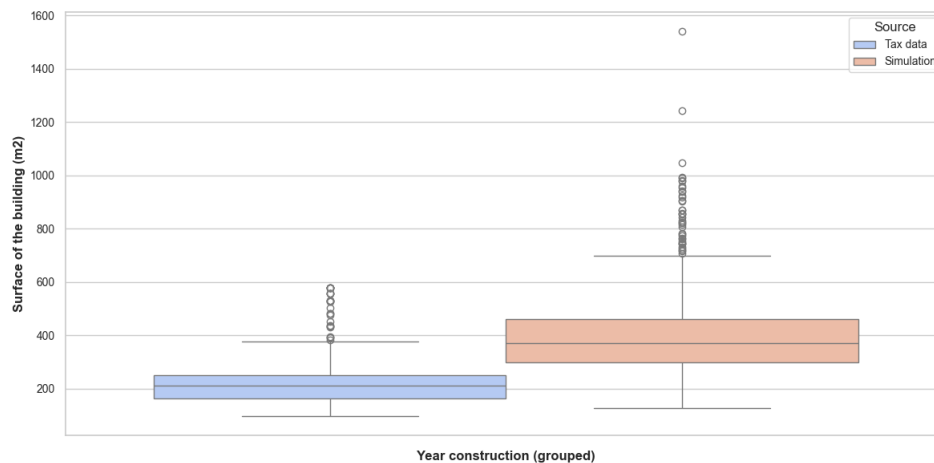
Figure 7. Comparison surface between tax office datasets and geojson extrapolated surface.

Given the fact that the methodology developed by the CERC uses real geometry, including shadings and adjacencies, scaling down these geometries without losing the geometrical links, the CERC team decided to incorporate a downsizing factor based on the value of the real surface from the tax evaluation data and apply the factor between the surface tax data and the simulated surface data to the final energy results. Regarding energy consumption, as described in the methodology, the simulation data were compared with a set of data coming from Energuide audits. More than 19.111 values were filtered for the typology of simulated buildings located in Montréal. The values for the Energuide cases were compared in vintage groups with the simulated cases in Figure 8. Although some vintages had relatively limited data, the values obtained with the simulation were very well aligned in most cases. Only the vintage between 1978 and 1983 showed significantly different values, but the limited number of buildings simulated for this vintage (4) made this less significant.
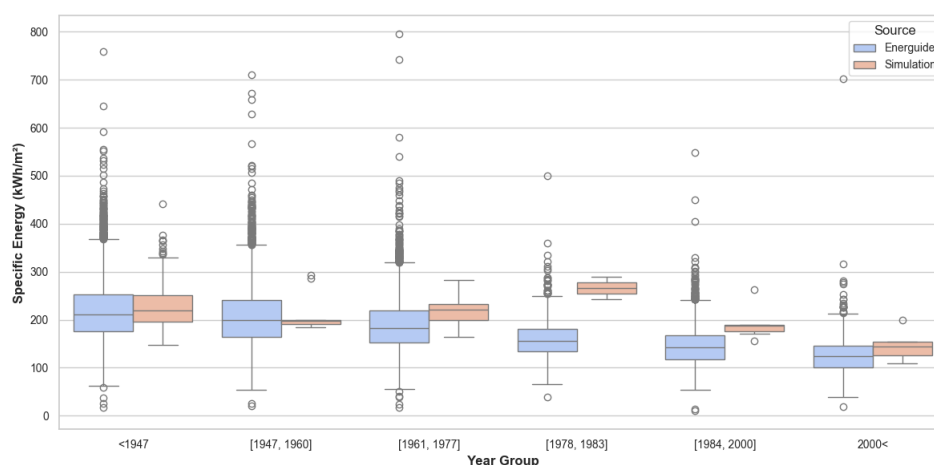

Figure 8. Final specific energy in single-family/duplex/triplex buildings from Energuide dataset compared to simulation data.

4.3.2. Institutional Buildings

For the tertiary sector buildings, we analyzed the list of buildings published by the City of Montreal and their energy consumption. We evaluated 15 of these buildings regarding the surface

area and consumption in Figure 9. In terms of surface area, the results were uneven. In five cases, the differences were more than 30% of the surface. Overall, using the 3D models extracted from the city's data gave us 30% more surface area than the data published by the City. The reason behind this seemed to be the same as in the residential building case and was linked to a low level of detail in the current 3D data sources (currently between LoD1 and LoD2); this could be improved if the city agents developed better and more coherent geospatial models.
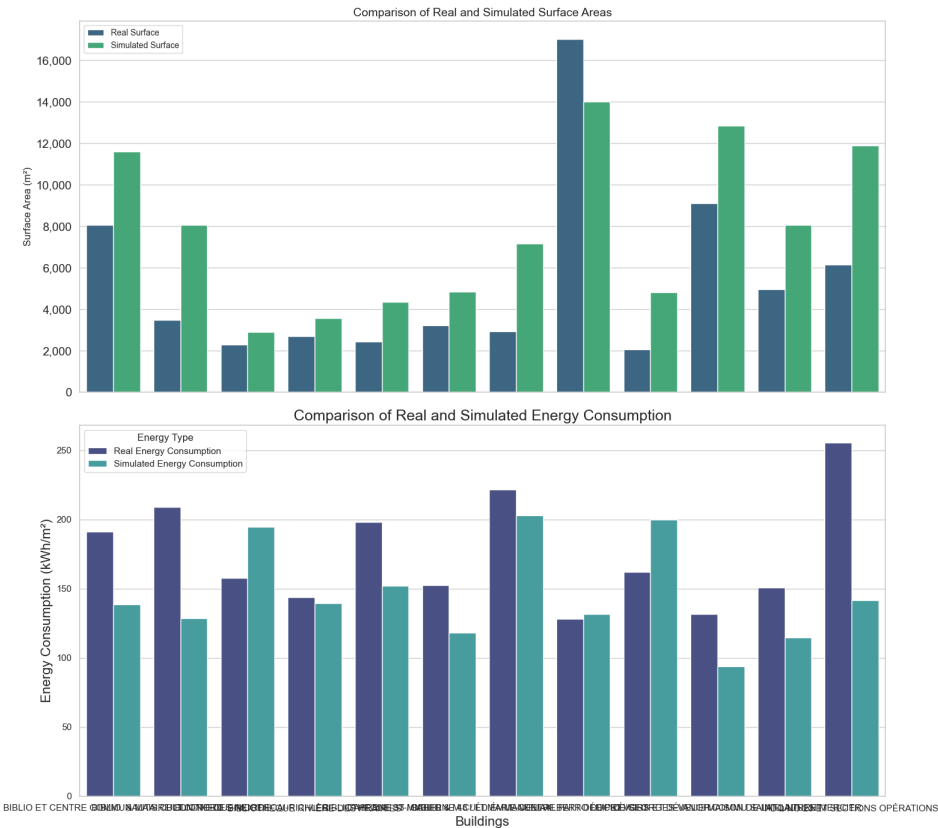
Figure 9. Comparison of surface of institutional buildings between city source and extracted data.

Concerning the energy consumption (final energy), we compared the results with the municipality's list. We noted that the difference was significant (more than 20% of the base case consumption) in three cases, but we aligned with the magnitude of specific energy consumption. When we did not match the results, the difference could be attributed to indoor environmental quality issues or other discrepancies from the assumed operating conditions. The impact of using different weather data files was analyzed (HQ data for the MacTavish weather station in Montreal compared with the Energyplus file used in the simulations [64]), and the difference between using one or the other was less than 5%. The results from the validation for individual buildings showed that the tool can also be used to find inconsistencies between the different datasets. The fact of having a tool that can help fill the gaps and detect anomalies in consumption and geometrical data seems an exciting capability, complementary to the tasks that are being performed using Machine Learning (ML) techniques in other areas of geospatial data treatment [65].

4.3.3. Energy Impact of Retrofitting Scenarios in Single-Family Buildings

Once the methodology had been validated, we tested the entire workflow, analyzing the impact of the three retrofit scenarios on the pilot cases. For the single-family/duplex areas, the impact of the three retrofitting scenarios was analyzed, and the workflow automatically updated the different parameters and assigned different costs to the scenarios. The results are shown in Figures 10 and 11. Regarding energy consumption, we can see that the impact of retrofitting was more critical in smaller and older

buildings, although there was no complete homogeneity. The fact that some buildings have adjacencies and, therefore, retrofitting would affect them less might have an influence. Standalone buildings would benefit more from retrofitting. The results from the tool showed that, even if some trends were clear and there was no need for a detailed analysis, the real impact of the measures for the different typologies and locations of these buildings has to be analyzed for more focused policy-making.
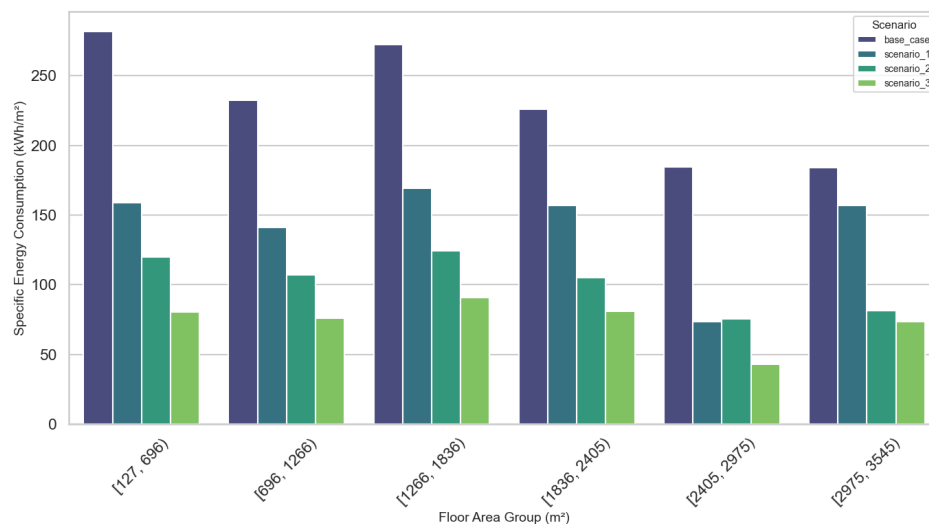

Figure 10. Specific final energy consumption per floor area group for the two neighborhoods.
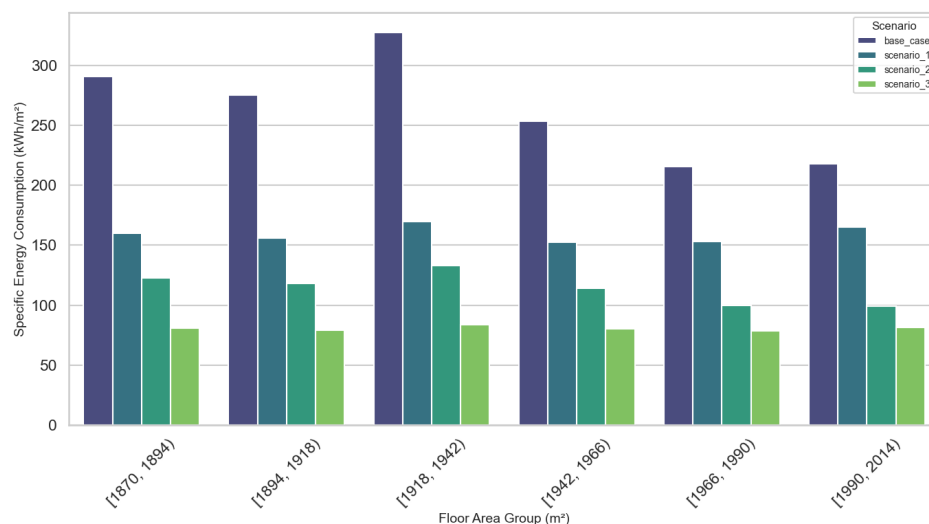
Figure 11. Specific final energy consumption per vintage group for the two neighborhoods.

4.3.4. Energy Impact of Retrofitting Scenarios on Institutional Buildings

In the case of the institutional buildings, the trends were more complicated to detect, given the small number of buildings and the different uses between them. Nevertheless, within the different building sets, we can see that a tool such as ours can be used to prioritize investments in specific typologies of municipal buildings and to see which scenario would be more beneficial, depending on the different buildings.

4.4. Economic Results

Once the energy consumption for each building in Montréal had been calculated, the retrofit cost workflow was implemented with the following economic parameters:

- number of years = 40,

- percentage credit = 0,
- interest rate = 0.04,
- consumer price index = 0.04,
- electricity peak index = 0.05,
- CO_2 price = 30 $/ton ,
- CO_2 price index = 0.06 (increase in CO2 price),
- electricity price index = 0.05,
- gas price index = 0.05,
- discount rate = 0.03 [66],
- retrofitting year construction = 2020,

The number of years for calculating the life cycle costs was established as 40 years. This decision was taken because, although the [66] used 30 years, the new proposed EU [67] aims at calculations using 50 years. The midpoint of 40 years was chosen.

We used the same sets of buildings in the validation phase to evaluate the economic impact of the different measures for the final set of buildings. Although this part of the process was not cross-validated with any actual projects, references such as [68,69] helped us see if we were aligned with the real economic costs of retrofitting.

Small Residential Buildings

The two residential neighborhoods helped us understand the economic impact of retrofitting with the previously established hypotheses. The results for the life cycle costs for both areas are plotted in Figures 12 and 13.
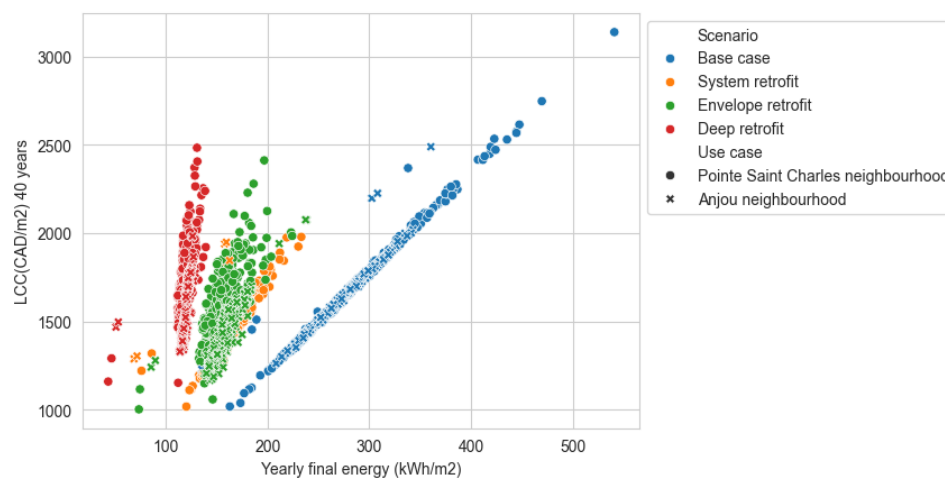

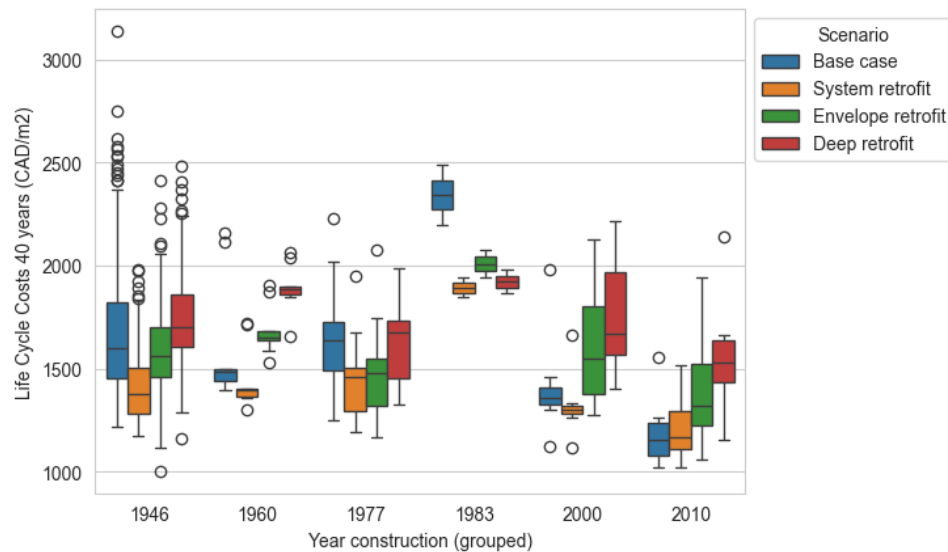**Figure 12.** Scatter plot for the different scenarios divided by retrofit scenario and neighborhood.

Figure 13. Box plot of costs per categories and per vintage for different scenarios.

We can observe that from a life cycle perspective, retrofitting the building brought solutions that, at least, have the same costs as the base case. The impact of the retrofitting was different depending on the vintage of the building.

We observed that skin retrofitting was interesting for buildings built before 2000. An energy systems retrofit alone was interesting depending on the vintage; with newer buildings being, generally, more interesting from an energy systems perspective. As a last point, deep retrofits did not seem to be economically feasible, even from a life cycle perspective.

Tertiary Buildings

In the case of tertiary buildings, Figure 14 shows a scatter plot of the life cycle costs of the buildings under the different scenarios; given the small size, no box plot depending on vintage was produced.
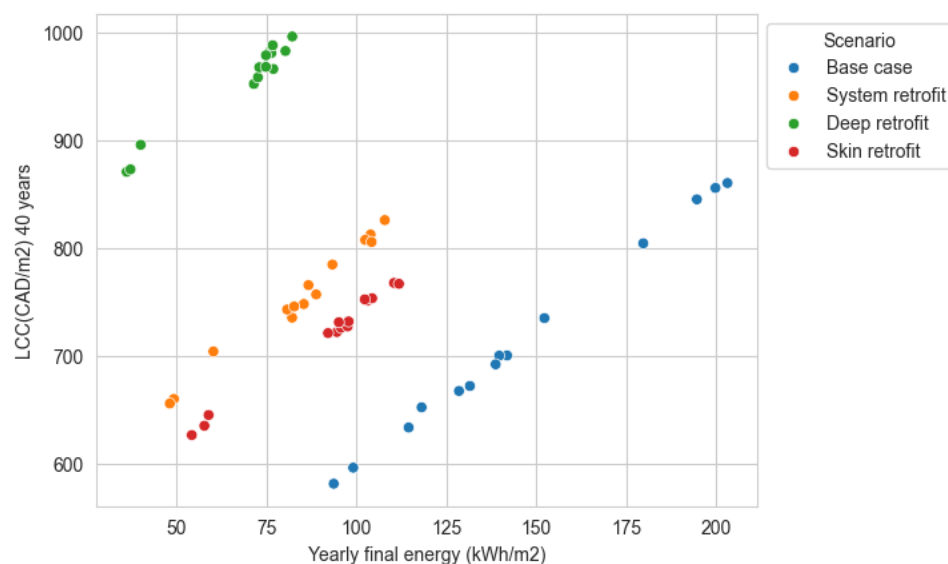

Figure 14. Scatter plot for the individual tertiary buildings.

In the case of tertiary buildings, the life cycle costs of the base case were lower than any of the retrofitting scenarios. The low electricity prices in large buildings in Montréal (under 0.07 CAD/kWh on average) make it very difficult to implement economically feasible energy efficiency solutions. The

results show that the values mentioned above in the literature are coherent with the CAPEX and OPEX of similar building types [70].

5. Research Limitations

Despite the efforts expended in the development of the tool, there are still some aspects that need to be improved:

- Data quality: The simulation results heavily depend on the quality of the geometry data used to create the 'City'. Currently, only LOD1 geojson files can be used for running EnergyPlus simulations, which should be improved soon.
- Lack of data: In addition to data quality, the existence of data for all different types of buildings is a big challenge. Moreover, finding compatible datasets for construction and usage based on the building function is another problem. Moreover, as mentioned in the text, there are not enough data about the existing energy systems in buildings, which affected the base case simulations heavily.
- Multi-zoning: Currently, all buildings are simulated as a single zone. This is one of the tool's limitations, which could affect the simulation results.
- Occupancy modeling: The utilization of fixed schedules for modeling occupancy is another tool limitation. A stochastic occupancy modeling workflow is currently under development.
- Economic Analysis: Life cycle costing (LCC) analysis uses generalized economic parameters, which might not be representative for broader applications or specific local economic conditions.

6. Conclusions and Next Steps

We presented a tool that can fully automate the simulation of large city areas (even in a large city such as Montréal) using a white-box approach and filling gaps related to low-quality input datasets. The process of validation of the results was generally satisfactory. However, the next stage of parameter calibration will be initiated in the following months, to understand if the discrepancies between the simulated data and real/statistical data could be adjusted by changing specific parameters.

This tool uses an open-source modular approach that allows the integration of the work of other researchers on the topic and the analysis of future building scenarios considering cross-sectoral criteria.

In conclusion, the TOOLS4Cities hub enables rapid analysis of the costs and carbon emissions of every building in a city. Based on archetypes for different building uses, the models allow comparing different renovation options. In this way, we hope to support and accelerate actions to decarbonize the municipal built environment.

7. Future Steps

The goal for the UBEM that we have developed, based on open-source criteria, modularity, flexibility, and integrability, is that it never stops evolving in the coming years.

The development goal is to make the framework available to the full research community and allow the integration of new workflows. Although any researcher can use the repository from the CERC, clone it, and use it, we encourage using the manuals and the hub methodology to increase the tool's capabilities.

The scalability of the tool is being evaluated at the time of the publication of this paper, with new pilot cases being developed in Palma de Mallorca (Spain), Québec City (QC, Canada), and Toronto (ON, Canada).

Data Availability Statement: All code and data used in this project are available in the web page of the Next Generation Cities Institute. Please contact the authors to gain access to the necessary folders and the open-source tools from the NGCI. The gitea with the source code for the TOOLS4Cities hub is in https://ngci.encs.concordia.ca/gitea/CERC/hub.git; accessed on 25 January 2025 and the costs library is in https://ngci.encs.concordia.ca/gitea/CERC/costs.git; accessed on 20 January 2025. The CKAN is in https://nextgenerations-cities.encs.concordia.ca/ckan/; accessed on 20 January 2025

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence. 15
BTAP	Building Technology Assessment Platform. 14, 15
CAPEX	Capital Expenditures. 12
CEA	City Energy Analyst. 3
CERC	Canada Excellence Research Chair for Sustainable Cities and Communities. 15, 16
CityGML	City Geography Markup Language. 2, 3
EnergyADE	Energy Application Domain Extension. 3
GHG	GreenHouse Gases. 1, 6
GIS	Geographic Information System. 6
HVAC	Heating, Ventilation, and Air Conditioning. 2, 11, 15, 28
IoT	Internet of the Things. 28
LCC	Life Cycle Costing. 1, 11
LiDAR	Laser Imaging Detection and Ranging. 14
LoD	Level of Detail. 3, 12, 14
ML	Machine Learning. 15, 18
MVP	Minimum Value Product. 7
NECB	National Energy Code for Buildings Canada. 15
NRCAN	Natural Resources Canada. 14, 15
NREL	National Renewable Energy Laboratory. 3, 14
PV	Solar Photovoltaics. 11
RC	Resistance Capacitance. 3
ROM	Reduced Order Models. 2
SRA	Simplified Radiosity Algorithm. 2
TEASER	Tool for Energy Analysis and Simulation for Efficient Retrofit. 3
UBEM	Urban Building Energy Modeling. 2, 3, 4, 5, 6, 8, 22, 27, 28
UHI	Urban Heat Island. 2
UMI	Urban Modeling Interface. 3
URBANOPT	Urban Renewable Building and Neighborhood Optimization. 3

Appendix A. Software Perspective

Delving deeper into the details of each software component will allow a more detailed understanding of the full architecture of the development used to create the data structure that lies at the core of the UBEM.

Appendix A.1. Models

A model is specific to a particular domain (e.g., transport or waste or building physics, etc.) and focuses on representing a single system (or a small number of closely related systems) within that

domain. A model is an algorithmic representation of an urban system capable of making predictions about the system's behavior or properties based on a definable set of inputs.

The TOOLS4Cities platform architecture is designed to integrate with any model, provided it can run on a computer. In this way, the platform has no theoretical restrictions on the scope of urban phenomena it can simulate, and it is also infinitely extendable over time as more models are added. Therefore, the platform uses 'model' solely as defined above and will use the full names of these other models to differentiate them. These models do not necessarily need to be developed with the same structure as the rest of the code developed by the CERC team, but they can be adapted to match the inputs and outputs required by the platform. However, they must be 'free-to-use' to be incorporated seamlessly into the platform. As such, they do not form part of the coding environment of the platform; instead, they are separate programs that must be accessible to the platform via their API (application programming interface). The requirements established by the CERC platform, as with any model used by the platform, are that all third Party Models must support the following:

- Use a definable and program-accessible set of input data
- Algorithms or calculations that use the input data.
- Produce a definable and program-accessible set of output data.
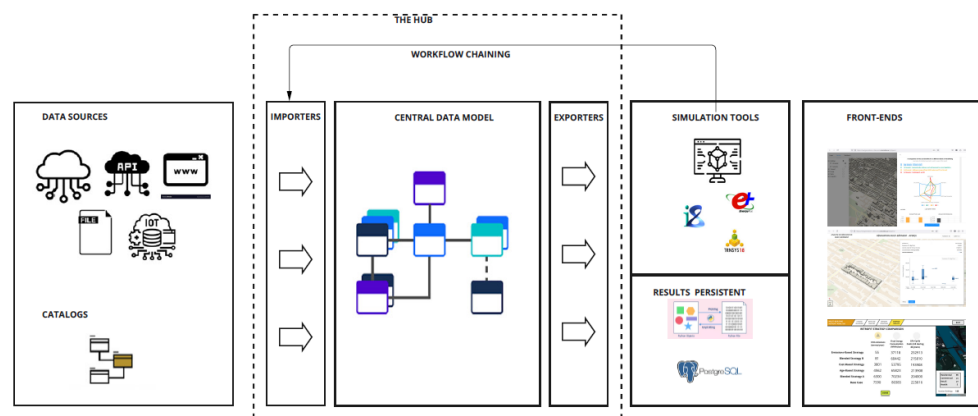

Figure A1. General view of the platform.

Appendix A.2. Central Data Model

A data model is a formalized structure that shows how data are to be represented within a system. Importantly, the data model is separate from the data and says nothing about where the data are physically stored. A data model is an abstract structure that allows us to understand the relationships between all pieces of data in a system.

Appendix A.3. Data Models

At the heart of the platform are the data models. Given all the aforementioned gaps in the existing UBEM models, it appears that it is a clear necessity to structure all the data. This data structure will be used to create clear interactions between the different sectors of urban decarbonization use cases and, at the same time, to use UBEM tools as multi-faceted tools and adapt the final tool at any moment.

For this reason, the authors developed a clear structure of how a city should be developed in Python classes. Each class (see UML diagram) has clear relationships with the others and can propose an association, multiplicity, aggregation, composition, inheritance, or realization [71]. The objective is to create a metamodel of the full city, with clear relationships between the essential parameters of each element that composes it and the relationships between its agents.

Developing a generic metamodel of building data (especially energy data) has been an objective of several initiatives in recent years, especially for energy utilities and the HVAC IoT sector. IFC, Brick, Haystack, and Bigg [72,73] are examples of metamodels and ontologies that have tried to structure

and organize the metadata from buildings to be able to understand their behavior and, in most cases, act on it.

A building-level perspective, though, is too detailed, and although some of the members of the same team have been developing a metamodel [72], the goal of the platform is to capture less data but remain compatible with the previously developed model.

The central data model consists, essentially, of a class called city that contains several components that will be stored in a structured manner. Buildings, roads, networks, public spaces, and waste separation plants are some of the objects that compose the city. Since the present article focuses on developing an urban retrofit model, a more detailed description of the building class will be developed.

The building class contains all the information necessary to develop global decarbonization analyses on buildings. The structure of internal zones, thermal zones, and thermal boundaries (surfaces, thermal openings, layers,..) contains all the input metadata that will allow the export to tools that can develop analyses such as energy consumption calculation, life cycle analysis, energy retrofitting analyses, demand response strategies (at urban level), and others. The flexible construction of the classes, with an orientation more similar to an ontology than a fixed metamodel, and with links and relationships that can be changed, allows for future code re-structuring. Using classes in Python instead of fixed XML structures allows for the fast use of links and a flexibility of connections.

A data model can have classes with duplicated properties, but the central data model uses the concept of inheritance to remove this duplication. The benefit of this is massively reducing the coding effort, making the code more reusable and easier to read. It also explicitly defines a relatedness between entities that helps the data model reflect the real interactions.

It is essential to underline that classes are only templates that guide how to represent the data of different entities. Once the classes have been called, the values that fill the templates are objects that are instantiated, reserving RAM memory and initializing the property values. These objects disappear once the process is finished. Therefore, the goal of the central data model is not to produce anything, but to give adequate structure for the later usage of the captured values.

Appendix A.4. Data

In the case of the Platform, the data used by the different workflows and models are divided between catalogs, datasets prepared by the CERC group, and online available data. The strategy of the CERC is to access, as much as possible, open data sources and, using the factories, link them to the central data model. Whenever a source of data requires deeper development or the origins of the data sources are unclear, a catalog is developed. The XML format is encouraged because of its clear structure and access facility. However, when the sources are well documented and maintained, external sources are directly accessed by the platform code. This strategy ensures that the amount of catalogs and data to be maintained is minimized and relies on the maintenance of these data sources by the external entities that are the data stewards. For example, the GitHub repository of the BTAP tool from NRCAN is used [74].
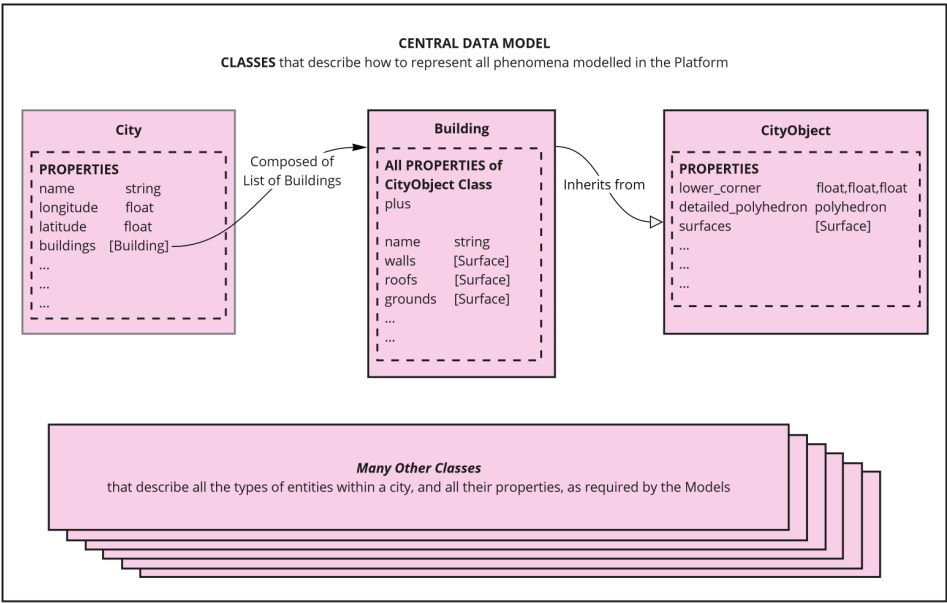
Central Data Model

Figure A2. Characteristics of the central data model.

Appendix A.5. Factories

As described in Figure A4, there are two blocks with coding components called 'Import Factories' and 'Export Factories'. Factories are essential, and most of the authors' efforts to achieve homogenization of different datasets came from those factories. The Factories are specialized functions that manage the exchange of data between various external (non-platform) data representations and the platform's internal central data model.
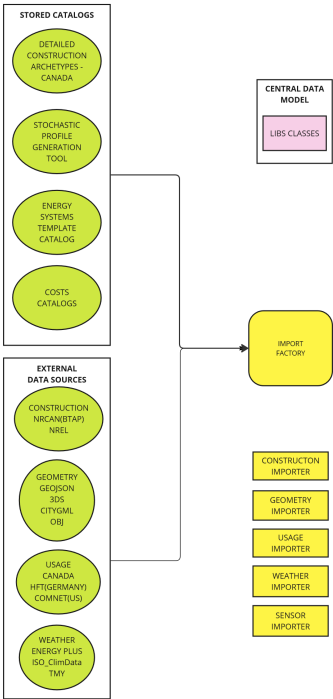

Figure A3. Detail on the platform workflow regarding input datasets.

When data are converted into the central data model representation, they are said to be 'imported'. When data are taken out of the central data model into an external representation, they are said to be 'exported'. All factories need a handler to know which type of format is being imported, as several types are supported by each factory.
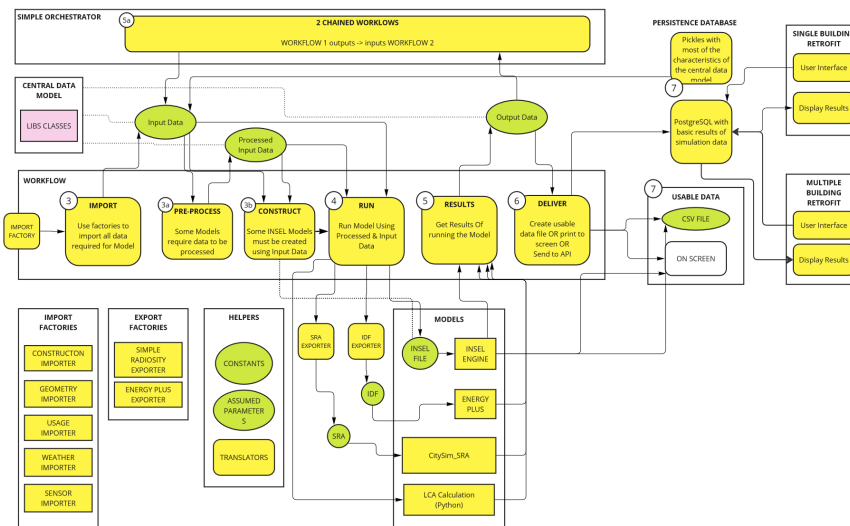

Figure A4. Steps of a complete workflow using the platform.

Appendix A.5.1. Import Factories

At the time of writing, there are import factories to handle the following categories of data: construction, geometry, weather, building usage, and sensors. The import factory for each data category can then handle various data formats. For example, the geometry importer can import data in both CityGML format, Geospatial JavaScript Object (GeoJSON) format, STL format, 3DS (Rhino), and OBJ format. The same platform includes the capacity to convert one object type into another. In this case, the handler of the Geometry factory supports at least cityGML, OBJ, rhino, and geojson. If a geojson file is imported, we must provide the handler geojson. Weather importers incorporate how to enrich a city object with weather data (from an EPW or a TMY file). The construction factory imports construction characteristics for each of the buildings of a city class, which are structured in construction archetypes. The construction handlers available at the moment are NREL, using the data from the DOE archetypes; and NRCAN, using BTAP standards, The usage factory, at the moment, enriches each of the buildings with the usage associated with each of the building functions/archetypes and has three handlers: one using the COMNET values and profiles, another using the data from the National Energy Code of Canada, and the third using a stochastic profile generator developed by one of the researchers of the team.

Appendix A.5.2. Export Factories

These do the same thing in reverse, taking some category of the data represented within a city object and converting it into an externally recognized format. This externally recognized format will be used in the different models to run use cases. At the moment, several handlers have been developed in the hub and more are being developed. The stable handlers, at the moment, include the Energyplus handler, the Monthly Energy Balance handler, and the SRA handler. The handlers being built are an RC model handler based on an R1C1 model (CityBEM) and the TRNSYS handler.

Appendix A.5.3. Helpers

Helpers are considered part of the factories and are the tools used to perform several functions in support of the homogenization of the central data model parameters. Helpers are used to define assumptions for models, declare constants, and perform data translations.

Development of Workflows

Once the pieces of the structure have been developed, the composition between its parts is defined. So far, we have considered a generalized Workflow and noted how its primary role is to manage the flow of data from external sources to one or more models, which it runs and then collects the resulting output data.

Looking at more details, workflows need to achieve several specialized things to carry out this role. The schema above includes more workflow components, explaining the different workflow relationships. The schema also shows a series of numbered steps. These are the fundamental steps that every workflow needs to take, regardless of which model it is managing. The different steps followed by workflows are as follows:

Step 1—IMPORT

Every workflow must first retrieve all the required data as input for the model(s) it manages. Data sources may be in proprietary formats (for example, building geometry may be in CityGML or OBJ format, while weather information may be in EPW, TMY2, or ISO CLIM format). Whenever a new model is integrated into the platform, we also develop new factory importers if data formats are not already supported. The import step is completed when the imported data are converted into a format that all other workflows in the platform can understand. This format is known as the central data model and has been explained more fully previously. Suffice to say, once the IMPORT step is complete, all the data from external sources are fully represented internally in the platform.

Step 1a—PRE-PROCESS (Optional)

Some models require that the input data retrieved from external data sources be altered in some way, before they are ready to be used by the model. There are many examples of this: 'Data Cleaning' (detecting and removing erroneous values; 'Data Calibrating'(modifying values based on auxiliary reference data); 'Data Pruning' (removing unnecessary data branches); 'Data Compression' (making the size of data more manageable). We call these steps 'Pre-Processing' and the workflow is responsible for carrying these out across all of the input data, as required.

Step 1b—CONSTRUCT (Optional)

For some models, the input data can change the model. For example, an INSEL model for calculating building energy demand may depend on the building geometry. The connectedness of thermal zones and surfaces within a building directly impacts the logic connections in a building simulation model. As a result, the input data must be analyzed and used to create the required building simulation model (INSEL, Energyplus, or TRNSYS). This is achieved programmatically by the CONSTRUCT step of the workflow.

Step 2—RUN

Once the workflow has the input data ready, it is time to RUN the model(s). This could be as simple as making one call to the model with one dataset, or it could involve calling the model multiple times with subsets of the data. For example, the input data may include 300 buildings in a city district, and the model can calculate them all, either using a multi-zone approach to calculate different buildings or running all the buildings in a model one by one. If the model is a custom model, written in Python, the input data can be sent using the format described by the central data model. However, if the model is an INSEL model, the input data need to be sent using an INSEL file template. And if the model is a third party model (e.g., Energy Plus), the input data must be sent in the proprietary format(s) understood by that model. (e.g., IDF files for Energy Plus). As with importing different data formats, the platform helps workflows here, providing factories that can easily export data to the format required. Again, whenever a new model is integrated into the platform, we also develop new factory exporters if any data formats are not already supported. The RUN step is complete once the workflow has called the model across all the necessary input data.

Step 3—RESULTS

All models generate some results or output data. The workflow must collect these data so that they can be used. INSEL models can output their data directly to the screen. They can also write their data to files (CSV and other formats). Energy Plus outputs text files. Custom models may do any of the above or return their results programmatically. The workflow must retrieve the output data from any of these sources, using factory importers again where necessary, and convert the output data once more into the format described by the central data model. This allows the data to be accessed by other models later in the same workflow or even other workflows after this is completed. When the workflow calls a model multiple times with different subsets of input data, the workflow must also retrieve the results after each call, building up a full set of output data.

Step 4—DELIVER

With the output data now retrieved and stored, the final responsibility of the workflow is to make the data available to whoever needs them. There are multiple options here. The Workflow may write the data to output files, display the data on the screen, and ensure the data are available for new Workflows; it may ensure the data are available for third party apps. In future versions of the platform, this part of the workflow is expected to be more formalized. Figure A4 shows the relationship between the central data model and the factories. The next two sections explain how these both support the workflow.

Appendix B. Detailed Hypotheses of the Archetypes

Appendix B lists the hypotheses used for construction materials for the different buildings from the different data sources compiled by the authors listed in the reference section.

Table A1. Composition of the walls. Archetypes for residential buildings.

Area	Opaque Walls	Transparent Walls
Pre-1950	Wall with 10 cm brick, 10 cm LW concrete, 10 cm air gap, 1.2 cm plasterboard. U = 1.498 W/m2K	Window with a glazing conductivity value of U = 3.10 W/m2K, a marc conductivity of U = 4.20 W/m2K, and an SHGC of 0.66.
	Roof membrane, insulation to achieve U = 0.823 W/m2K, metal surface	
	Floor with insulation to U = 0.678 W/m2K, 4-inch concrete, carpeting	
1950–1980	Wall with 10 cm brick, 10 cm LW concrete, 5 cm insulation, 10 cm air gap, 1.2 cm plasterboard.	Window with a glazing conductivity value of U = 3.10 W/m2K, a marc conductivity of U = 4.20 W/m2K and an SHGC of 0.66.
	Roof membrane, insulation reaching U = 0.823 W/m2K, metal surface	
	Floor with insulation to U = 0.678 W/m2K, 4-inch concrete, carpeting	
1980–2010	Wall with 25 mm stucco, 5/8″ plaster, virtual insulation to achieve U = 0.426 W/m2K	Window with a glazing conductivity value of U = 2.8 W/m2K, a marc conductivity of U = 4.20 W/m2K and an SHGC of 0.66.
	Roof membrane, insulation reaching U = 0.276 W/m2K, metal surface	
	Floor with insulation to achieve U = 0.459 W/m2K, 4-inch concrete, carpet	
2011-2020	Wall with 25 mm stucco, 5/8″ plaster, virtual insulation to achieve U = 0.247 W/m2K	Window with a glazing conductivity value of U= 2.2 W/m2K, a frame conductivity of U = 3.1 W/m2K and an SHGC of 0.39.
	Roof membrane, insulation reaching U = 0.183 W/m2K, metal surface	

Table A1. *Cont.*

Area	Opaque Walls	Transparent Walls
	Floor with insulation to reach U = 0.183 W/m2K, 4-inch concrete, carpet	
>2020	Wall with 25 mm stucco, 5/8″ plaster, virtual insulation to achieve U = 0.247 W/m2K	Window with a glazing conductivity value of U = 1.9 W/m2K, a frame conductivity of U = 2.20 W/m2K and an SHGC of 0.39.
	Roof membrane, insulation reaching U = 0.138 W/m2K, metal surface	
	Floor with insulation to reach U = 0.156 W/m2K, 4-inch concrete, carpet	

Table A2. Composition of the walls. Archetypes for non-residential buildings.

Area	Opaque Walls	Transparent Walls
Pre-1950	Brick/Stone/Terracotta/Concrete with an overall U value of U = 0.9 W/m2K	Window with a glazing conductivity value of U = 5 W/m2K, a frame conductivity of U = 4.20 W/m2K, and an SHGC of 0.8.
	Roof membrane, insulation to achieve U = 0.823 W/m2K, metal surface	
	Floor with insulation to U = 0.678 W/m2K, 4-inch concrete, carpeting	
1950–1980	Steel structure/Curtain wall, Brick/Stone Cladding: 0.1, Concrete: 0.1, Gypsum Plastering: 0.013.	Window with a glazing conductivity value of U = 3.10 W/m2K, a frame conductivity of U = 4.20 W/m2K and an SHGC of 0.66.
	Roof membrane: 0.002 m, Asphalt cover board: 0.01 m, Rigid insulation (e.g., MW Glass Wool): 0.10 m–0.12 m, Steel trusses, joists, concrete decks, parallel-chord trusses and joists Floor with insulation to U = 0.678 W/m2K, 4-inch concrete, carpeting	
1980-2010	Curtain wall with veneer or precast cladding: 0.1 m, Insulation (e.g., Rigid board insulation): 0.10 m–0.12 m, Interior gypsum plastering: 0.013 m./Metallic Cladding: 0.006 m, Gypsum Plastering: 0.013 m, Insulation (e.g., Rigid board insulation): 0.10 m–0.12 m. U = 0.426 W/m2K	Window with a glazing conductivity value of U = 2.8 W/m2K, a frame conductivity of U = 4.20 W/m2K and an SHGC of 0.66.
	Roof membrane: 0.002 m, Asphalt cover board: 0.01 m, Rigid insulation (e.g., MW Glass Wool): 0.10 m–0.12 m, Steel trusses, joists, concrete decks, parallel-chord trusses and joists U = 0.276 W/m2K	
	Floor with insulation to achieve U = 0.459 W/m2K, 4-inch concrete, carpet	
2011–2020	Brick veneer, with air space and insulation on steel or wood framing. U = 0.247 W/m2K	Window with a glazing conductivity value of U = 2.2 W/m2K, a frame conductivity of U = 3.1 W/m2K and an SHGC of 0.39.
	Roof membrane, insulation reaching U = 0.183 W/m2K, metal surface	
	Floor with insulation to reach U = 0.183 W/m2K, 4-inch concrete, carpet	
>2020	Brick veneer, with air space and insulation on steel or wood framing. U = 0.247 W/m2K U = 0.247 W/m2K	Window with a glazing conductivity value of U = 1.9 W/m2K, a frame conductivity of U = 2.20 W/m2K and an SHGC of 0.39.
	Roof membrane, insulation reaching U = 0.138 W/m2K, metal surface	
	Floor with insulation to reach U = 0.156 W/m2K, 4-inch concrete, carpet	

References

1. Programme, U.N.E.; Construction, G.A.f.B. *2020 Global Status Report for Buildings and Construction: Towards a Zero-Emissions, Efficient and Resilient Buildings and Construction Sector—Executive Summary*; United Nations Environmental Program, Nairobi 2020.

2. Ding, G. Demolish or refurbish—Environmental benefits of housing conservation. *Australas. J. Constr. Econ. Build.* **2013**, *13*, 18–34. https://doi.org/10.5130/ajceb.v13i2.3322.

3. Power, A. Does demolition or refurbishment of old and inefficient homes help to increase our environmental, social and economic viability? *Energy Policy* **2008**, *36*, 4487–4501. https://doi.org/10.1016/j.enpol.2008.09.022.

4. Schwartz, Y.; Raslan, R.; Mumovic, D. Refurbish or replace? The Life Cycle Carbon Footprint and Life Cycle Cost of Refurbished and New Residential Archetype Buildings in London. *Energy* **2022**, *248*, 123585. https://doi.org/10.1016/j.energy.2022.123585.

5. Crawford, K.; Johnson, C.; Davies, F.; Joo, S.; Bell, S. Demolition or Refurbishment of Social Housing? In *Technical Report, UCL Urban Lab and Engineering Exchange for Just Space and the London Tenants Federation*; UCL, London, 2014.

6. Tetteh, M.O.; Boateng, E.B.; Darko, A.; Chan, A.P.C. What Are the General Public's Needs, Concerns and Views about Energy Efficiency Retrofitting of Existing Building Stock? A Sentiment Analysis of Social Media Data. *Energy Build.* **2023**, *301*, 113721. https://doi.org/10.1016/j.enbuild.2023.113721.

7. Brendan Haley, R.T. *Canada's Climate Retrofit Mission*, Efficiency Canada, Ottawa, 2021.

8. Lohse, R.; Mørck, O.C.; Zhivov, A., Eds. *Deep Energy Retrofit—Case Studies: Business and Technical Concepts for Deep Energy Retrofit of Public Buildings, Energy in Buildings and Communities Programme, Annex 61, Subtask A*; Springer International Publishing: Cham, Switzerland, 2023. https://doi.org/10.1007/978-3-031-17517-6.

9. Volans. *The retrofit revolution. Pattern and APPG on Fair Banking*; Bankers for Net Zero,2021

10. Prabatha, T.; Hewage, K.; Karunathilake, H.; Sadiq, R. To Retrofit or Not? Making Energy Retrofit Decisions through Life Cycle Thinking for Canadian Residences. *Energy Build.* **2020**, *226*, 110393. https://doi.org/10.1016/j.enbuild.2020.110393.

11. Reinhart, C.F.; Cerezo Davila, C. Urban building energy modeling—A review of a nascent field. *Build. Environ.* **2016**, *97*, 196–202. https://doi.org/10.1016/j.buildenv.2015.12.001.

12. Li, Z.; Ma, J.; Tan, Y.; Guo, C.; Li, X. Combining physical approaches with deep learning techniques for urban building energy modeling: A comprehensive review and future research prospects. *Build. Environ.* **2023**, *246*, 110960. https://doi.org/10.1016/j.buildenv.2023.110960.

13. Dabirian, S.; Panchabikesan, K.; Eicker, U. Occupant-centric urban building energy modeling: Approaches, inputs, and data sources—A review. *Energy Build.* **2022**, *257*, 111809. https://doi.org/10.1016/j.enbuild.2021.111809.

14. Dahlström, L.; Broström, T.; Widén, J. Advancing urban building energy modelling through new model components and applications: A review. *Energy Build.* **2022**, *266*, 112099. https://doi.org/10.1016/j.enbuild.2022.112099.

15. Keirstead, J.; Jennings, M.; Sivakumar, A. A review of urban energy system models: Approaches, challenges and opportunities. *Renew. Sustain. Energy Rev.s* **2012**, *16*, 3847–3866. https://doi.org/10.1016/j.rser.2012.02.047.

16. Allegrini, J.; Orehounig, K.; Mavromatidis, G.; Ruesch, F.; Dorer, V.; Evins, R. A review of modelling approaches and tools for the simulation of district-scale energy systems. *Renew. Sustain. Energy Rev.* **2015**, *52*, 1391–1404. https://doi.org/10.1016/j.rser.2015.07.123.

17. Robinson, D.; Campbell, N.; Gaiser, W.; Kabel, K.; Le-Mouel, A.; Morel, N.; Page, J.; Stankovic, S.; Stone, A. SUNtool—A new modelling paradigm for simulating and optimising urban sustainability. *Sol. Energy* **2007**, *81*, 1196–1211. https://doi.org/10.1016/j.solener.2007.06.002.

18. Robinson, D.; Stone, A. Holistic radiation modelling with a fast simplified radiosity algorithm. In Proceedings of the Proceedings of Building Simulation 2005: 9th Conference of IBPSA, Montréal, Canada, 15–18 August 2005.

19. Horni, A et al. (eds.) 2016. The Multi-Agent Transport Simulation MATSim. London: Ubiquity Press. 2016 https://doi.org/10.5334/baw

20. Scartezzini, J.L.; Nouvel, R.; BRASSEL, K.H.; BRUSE, M.; Duminil, E.; Coors, V.; Eicker, U.; Robinson, D. *SimStadt, a New Workflow-Driven Urban Energy Simulation Platform for CityGML City Models*; In Proceedings, CISBAT, Lausanne, 2015.

21. Monien, D.; Strzalka, A.; Koukofikis, A.; Coors, V.; Eicker, U. Comparison of building modelling assumptions and methods for urban scale heat demand forecasting. *Future Cities Environ.* **2017**, *3*, 2. https://doi.org/10.1186/s40984-017-0025-7.

22. Agugiaro, G.; Joachim, B.; Cipriano, P.; Nouvel, R. The Energy Application Domain Extension for CityGML: Enhancing interoperability for urban energy simulations. *Open Geospat. Data Softw. Stand.* **2018**, *3*, 2. https://doi.org/10.1186/s40965-018-0042-y.

23. Weiler, V.; Harter, H.; Eicker, U. Life cycle assessment of buildings and city quarters comparing demolition and reconstruction with refurbishment. *Energy Build.* **2017**, *134*, 319–328. https://doi.org/10.1016/j.enbuild.2016.11.004.

24. Reinhart, C.; Dogan, T.; Jakubiec, J.A.; Rakha, T.; Sang, A. UMI—An urban simulation environment for building energy use, daylighting and walkability. In Proceedings of the 13th Conference of International Building Performance Simulation Association, Chambery, France, 26–28 August 2013.

25. Wu, W.; Dong, B.; Wang, Q.R.; Kong, M.; Yan, D.; An, J.; Liu, Y. A novel mobility-based approach to derive urban-scale building occupant profiles and analyze impacts on building energy consumption. *Appl. Energy* **2020**, *278*, 115656. https://doi.org/10.1016/j.apenergy.2020.115656.

26. Baetens, R.; De Coninck, R.; Jorissen, F.; Picard, D.; Helsen, L.; Saelens, D. OpenIDEAS—An Open Framework for integrated District Energy Simulations. In Proceedings of the 14th Conference of International Building Performance Simulation Association, Hyderabad, India, 7–9 December 2015. https://doi.org/10.26868/25222708.2015.2243.

27. Ferrando, M.; Causone, F.; Hong, T.; Chen, Y. Urban building energy modeling (UBEM) tools: A state-of-the-art review of bottom-up physics-based approaches. *Sustain. Cities Soc.* **2020**, *62*, 102408. https://doi.org/10.1016/j.scs.2020.102408.

28. El Kontar, R.; Polly, B.; Charan, T.; Fleming, K.; Moore, N.; Long, N.; Goldwasser, D. URBANopt: An Open-Source Software Development Kit for Community and Urban District Energy Modeling. In Proceedings of the 2020 Building Performance Analysis Conference & Simbuild, Chicago, IL, USA, 26–28 Septembe 2020.

29. Deng, Z.; Chen, Y.; Yang, J.; Causone, F. AutoBPS: A tool for urban building energy modeling to support energy efficiency improvement at city-scale. *Energy Build.* **2023**, *282*, 112794. https://doi.org/10.1016/j.enbuild.2023.112794.

30. Yang, J.; Zhang, Q.; Peng, C.; Chen, Y. AutoBPS-Prototype: A web-based toolkit to automatically generate prototype building energy models with customizable efficiency values in China. *Energy Build.* **2024**, *305*, 113880. https://doi.org/10.1016/j.enbuild.2023.113880.

31. Chen, Z.; Deng, Z.; Chong, A.; Chen, Y. AutoBPS-BIM: A toolkit to transfer BIM to BEM for load calculation and chiller design optimization. In *Proceedings of the Building Simulation*; Springer: Berlin/Heidelberg, Germany, 2023; Volume 16, pp. 1287–1298.

32. Rayegan, S.; Katal, A.; Wang, L.L.; Zmeureanu, R.; Eicker, U.; Mortezazadeh, M.; Tahmasebi, S. Modeling building energy self-sufficiency of using rooftop photovoltaics on an urban scale. *Energy Build.* **2024**, *324*, 114863. https://doi.org/10.1016/j.enbuild.2024.114863.

33. Wang, X.; Tian, S.; Ren, J.; Jin, X.; Zhou, X.; Shi, X. A novel resistance-capacitance model for evaluating urban building energy loads considering construction boundary heterogeneity. *Appl. Energy* **2024**, *361*, 122896. https://doi.org/10.1016/j.apenergy.2024.1.

34. Wang, X.; Zhang, X.; Zhu, S.; Ren, J.; Causone, F.; Ye, Y.; Jin, X.; Zhou, X.; Shi, X. A novel and efficient method for calculating beam shadows on exterior surfaces of buildings in dense urban contexts. *Build. Environ.* **2023**, *229*, 109937. https://doi.org/10.1016/j.buildenv.2022.109937.

35. Yan, D.; Xia, J.; Tang, W.; Song, F.; Zhang, X.; Jiang, Y. DeST—An integrated building simulation toolkit Part I: Fundamentals. In *Proceedings of the Building Simulation*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 1, pp. 95–110.

36. Zhang, X.; Xia, J.; Jiang, Z.; Huang, J.; Qin, R.; Zhang, Y.; Liu, Y.; Jiang, Y. DeST—An integrated building simulation toolkit Part II: Applications. *Build. Simul.* **2008**, *1*, 193–209. https://doi.org/10.1007/s12273-008-8124-x.

37. Kavgic, M.; Mavrogianni, A.; Mumovic, D.; Summerfield, A.; Stevanovic, Z.; Djurovic-Petrovic, M. A review of bottom-up building stock models for energy consumption in the residential sector. *Build. Environ.* **2010**, *45*, 1683–1697. https://doi.org/10.1016/j.buildenv.2010.01.021.

38. Shaikh, P.H.; Nor, N.B.M.; Nallagownden, P.; Elamvazuthi, I.; Ibrahim, T. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renew. Sustain. Energy Rev.* **2014**, *34*, 409–429. https://doi.org/10.1016/j.rser.2014.03.027.

39. Monsalvete Alvarez de Uribarri, P. *A Multiscale Framework for Predicting Distributed Renewable Thermal Energy Integration*; University of Nottingham: Nottingham, UK, 2020.

40. Huber, J.; Nytsch-Geusen, C. Development of modeling and simulation strategies for large-scale urban districts. In *Proceedings of the Building Simulation*; Sydney, 14-16 November 2011; pp. 1753–1760.

41. Bollinger, L.A.; Evins, R. Facilitating model reuse and integration in an urban energy simulation platform. *Procedia Comput. Sci.* **2015**, *51*, 2127–2136. https://doi.org/10.1016/j.procs.2015.05.484.

42. Nouvel, R.; Mastrucci, A.; Leopold, U.; Baume, O.; Coors, V.; Eicker, U. Combining GIS-based statistical and engineering urban heat consumption models: Towards a new framework for multi-scale policy support. *Energy Build.* **2015**, *107*, 204 – 212. https://doi.org/10.1016/j.enbuild.2015.08.021.

43. Xu, Z.; Coors, V. Application of system dynamics, GIS and 3D visualization in a study of residential sustainability. In Proceedings of the International Conference on Computational Science and Its Applications, Santander, Spain, 20–23 June 2011; pp. 300–314.

44. CERC. Repository gitea https://ngci.encs.concordia.ca/gitea/CERC/hub; accessed on 20 January 2025

45. Dabirian, S.; Saad, M.M.; Hussain, S.; Peyman, S.; Rahimi, N.; Monsalvete Alvarez U, P.; Yefi, P.; Eicker, U. Structuring heterogeneous urban data: A framework to develop the data model for energy simulation of cities. *Energy Build.* **2023**, *296*, 113376. https://doi.org/10.1016/j.enbuild.2023.113376.

46. Schumacher, J. INSEL Documentation. Available online: http://insel.eu/index.php?id=301 (accessed on 20 January 2025).

47. Bleyl, J.; Bareit, M.; Casas, M.; Chatterjee, S.; Coolen, J.; Hulshoff, A.; Lohse, R.; Mitchell, S.; Robertson, M.; Ürge Vorsatz, D. Office building deep energy retrofit: Life cycle cost benefit analyses using cash flow analysis and multiple benefits on project level. *Energy Effic.* **2019**, *12*, 1–19. https://doi.org/10.1007/s12053-018-9707-8.

48. *ISO 15686-5:2017*; Buildings and Constructed Assets—Service Life Planning. ISO: Geneva, Switzerland, 2017. Available online: https://www.iso.org/standard/61148.html (accessed on 20 January 2025).

49. Charette, R.P.; Marshall, H.E. *UNIFORMAT II Elemental Classification for Building Specifications, Cost Estimating, and Cost Analysis*; Technical Report NIST IR 6389; National Institute of Standards and Technology: Gaithersburg, MD, USA, 1999. https://doi.org/10.6028/NIST.IR.6389.

50. Canada, N.R. *EnerGuide Energy Efficiency Home Evaluations*; Natural Resources Canada: Ottawa, ON, Canada, 2018.

51. Consommation d'énergie et émissions de gaz à effet de serre des bâtiments municipaux de 2000 m² et plus—Site web des données ouvertes de la Ville de Montréal. Data accessible in https://donnees.montreal.ca/dataset/consommation-emissions-batiments-municipaux (accessed on 20 January 2025)

52. Unités D'évaluation Foncière—Site Web Des Données Ouvertes de la Ville de Montréal. Open data from tax evaluation of all the buildings in Montréal https://donnees.montreal.ca/dataset/unites-evaluation-fonciere (accessed on 20 January 2025)

53. LiDAR aérien 2015—Site web des données ouvertes de la Ville de Montréal. LIDAR data from Montréal accessible in https://donnees.montreal.ca/dataset/lidar-aerien-2015 (accessed on 20 January 2025)

54. Canada, N.R. Automatically Extracted Buildings—Prepackaged Shapefiles (download directory)—Open Government Portal. List with automatedly extracted buildings from all Canada. Available at https://open.canada.ca/data/en/dataset/7a5cda52-c7df-427f-9ced-26f19a8a64d6 (accessed on 20 January 2025)

55. Laval, U. Geoindex—Public. Web with geospatial data from Québec, developed by the Université Laval. https://geoapp.bibl.ulaval.ca/ (accessed on 20 January 2025)

56. NRCAN. *OpenStudio-Standards*; National Renewable Energy Laboratory: Applewood, CO, USA, 2023.

57. Charbonneau, M. Simulating the Effects of Enclosure Retrofits on Post-War High-Rise Apartment Buildings in Cold Climates. Ph.D. Thesis, University of Waterloo, Waterloo, ON, Canada, 2011.

58. Jadaa, D.A.; McArthur, J.J. *Development of Granular Archetypes for Canadian Existing Commercial Office and Multi-Unit Residential Buildings in Climate Zone*; Proceedings of eSim 2024: 13th Conference of IBPSA-Canada Edmonton, Alberta, 5-7th June 2024

59. Canada, G.B.C. Decarbonizing Canada's Large Buildings. Available online: https://www.cagbc.org/news-resources/research-and-reports/decarbonizing-canadas-large-buildings/ (accessed on 20 January 2025).

60. Canada, N.R.C. National Energy Code of Canada for Buildings. 2020. Available online: https://nrc.canada.ca/en/certifications-evaluations-standards/codes-canada/codes-canada-publications/national-energy-code-canada-buildings-2020 (accessed on 20 January 2025).

61. Government of Canada, N.R.C. *Energy Use in Canada: Publications*; Government of Canada, Natural Resources Canada: Ottawa, ON, Canada, 2011.

62. Data portal from cities. https://dataportalforcities.org/ (accessed on 20 January 2025)

63. Chowdhury, S. Rapid Building Feature Extraction and Geometry Formulation Using Machine Learning. In Proceedings of the Eleventh National Conference of IBPSA, Denver, CO, USA, 21–23 May 2024.

64. EnergyPlus weather data for Montréal https://energyplus.net/weather-location/north_and_central_america_wmo_region_4/CAN/PQ/CAN_PQ_Montreal.Intl.AP.716270_CWEC (accessed on 20 January 2025)

65. Hecht, R.; Danke, T.; Herold, H.; Hudson, P.; Munke, M.; Rieche, T. Colouring Cities: A Citizen Science Platform for Knowledge Production on the Building Stock - Potentials for Urban and Architectural History. In *Research and Education in Urban History in the Age of Digital Libraries*; Münster, S., Pattee, A., Kröber, C., Niebling, F., Eds.; Springer, Cham, Switzerland . 2023; pp. 145–164. https://doi.org/10.1007/978-3-031-38871-2_9.

66. Directive, E.E. EU Energy Efficiency Directive 2012/27/EU. *J. Ref.* **2012**, *50*, 1–56.

67. Mazzarella, L. *The Revised Energy Performance Buildings Directive (EPBD) in the Frame of 55 Program*; 2024.

68. Echeverria, A.; Palacios, J.; Davila, C.; Zheng, S. Quantifying the financial value of building decarbonization technology under uncertainty: Integrating energy modeling and investment analysis. *Energy Build.* **2023**, *297*, 113260. https://doi.org/10.1016/j.enbuild.2023.113260.

69. Zhivov, A.; Lohse, R. *Deep Energy Retrofit—A Guide for Decision Makers*; Springer, Cham, Switzerland, 2021. https://doi.org/10.1007/978-3-030-66211-0.

70. Kennedy, M.; Frappé-Sénéclauze, T.P. *Canada's Renovation Wave A Plan for Jobs and Climate*; Technical Report; The Pembina Institute: Vancouver, BC, Canada,2021.

71. Admin. *What Are the Six Types of Relationships in UML Class Diagrams*? Web page with the description of UML diagrams https://blog.visual-paradigm.com/what-are-the-six-types-of-relationships-in-uml-class-diagrams/ (accessed on 20 January 2025)

72. Yefi, P.; Menon, R.; Eicker, U.; Guéhéneuc, Y.G. MetamEnTh: A Object-Oriented Metamodel for IoT Systems in Buildings. *IEEE Internet Things J.* **2024**, *11*, 25818–25838. https://doi.org/10.1109/JIOT.2024.3373330.

73. Project, BIGG. *Biggproject/Ontology*; European Commission project dedicated to Ontologies https://www.bigg-project.eu/ (accessed on 20 January 2025)

74. *NREL/Openstudio-Standards*; git repository of the NREL https://github.com/NREL/openstudio-standards (accessed on 20 January 2025)