Technical University of Munich

Department of Civil, Geo and Environmental Engineering

Chair of Computational Modeling and Simulation

Interoperability of BIM-based Life-Cycle Energy Analysis in Early Design Stages

Master Thesis

for the Master of Science Degree in Civil Engineering

Author:	Lothar Kolbeck
Student Code:	[redacted]
1. Betreuer:	Prof. Dr.-Ing. André Borrmann
2. Betreuer:	Kasimir Forth
Date of Issue:	01.04.2020
Date of Submission:	29.10.2020

Abstract

In the design process, life cycle energy analysis (LCEA) supports the architect with predictions on the building's operational and embodied energy performance. The earlier these predictions can be retrieved, the more they can influence outstanding design decisions. To encourage the early application, building information modeling (BIM) may increase the accuracy of results and eliminate time-consuming manual procedures. Therefore, robust and fast data exchange between BIM-authoring tools and the software performing the energy analysis has to be established. Problems commonly occur at the geometric conversion of the BIM-model to a building energy model (BEM). For most residential and some non-residential buildings, the building envelope can be divided into zones according to empirical data in the post-processing. Then, the error-prone placement and conversion of space elements can be avoided. Through direct access to building element information, a flexible treatment of BIM-models with commonly encountered quality is possible. Applying this simplification, a workflow leveraging the frequent and didactic usage of LCEA throughout the design process is developed. Enabling real-time feedback, the workflow is prototypically implemented using the Revit API and the LCEA approach of Hollberg (2016) in CAALA (2020). For a user to understand and review the procedure, a visualization of the BEM in the gbXML format can be shown. Finally, the algorithmics of the prototype is validated and its practical functionality is confirmed by an architect.

Content

Figures

Tables

List of Abbreviations

AEC	Architecture, Engineering and Construction
API	Application Programming Interface
BEM	Building Energy Model
BIM	Building Information Modeling
BoQ	Bill of Quantities
BoMQ	Bill of Material Quantities
B-Rep	Boundary Representation
CSG	Constructive Solid Geometry
e.g.	For example
EE	Embodied Emissions
gbXML	Green Building XML
HVAC	Heating, Ventilation, Air and Cooling
IFC	Industry Foundation Classes
LCA	Life Cycle Assessment
LCC	Life Cycle Costing
LCI	Life Cycle Inventory
LCEA	Life Cycle Energy Analysis
MVD	Model View Definition
MEP	Mechanical Equipment

OE	Operational Energy
PEC	Primary Energy Consumption
PED	Primary Energy Demand
PLN	Format that is used by ArchiCAD (Nemetschek)
RVT	Format that is used in by Revit (Autodesk)
SB	Space Boundaries
UML	Unified Modeling Language

1 Introduction

1.1 Motivation

The building sector is responsible for a significant part of human environmental impact. It accounts for approximately 40 % of the world's primary energy demand and nearly one-third of global greenhouse gas emissions (Hegger et al. 2007). To downscale these measures, policy-makers introduced progressively stricter regulations on the operational energy demand of buildings. Due to the decrease in operational energy demand, the other life phases come into focus as relevant contributors to the energy performance of a building (Schöndube 2016). Therefore, the standardized method of life cycle energy analysis (LCEA) considers operational as well as embodied energy demand (Ramesh et al. 2010).

Applied during a design process, LCEA may support the practitioner with a prediction of a planning state's energy performance. The result should guide optimization through changes in geometry, materials, and mechanical equipment. These design changes require less effort and cost the earlier they are applied. Unfortunately, LCEA is commonly applied at late design stages because of the complexity and data demands of the software solutions available (Hollberg 2016). In order to incentivize the early and continuous use of the method, the LCEA method must operate in another context. This context differs in three aspects: First, the person responsible for the design is commonly not a specialized energy consultant, but rather an architect (Schlueter und Thesseling 2009). Second, the data required for the calculations is not fixed, but variable. Third, the effort must not overdemand the user, but enable frequent and didactic usage. Accounting for these requirements, building information modeling (BIM) is suitable as technological support. BIM refers to the creation and the access of a digital model as a shared knowledge resource (Borrmann et al. 2018). This BIM-model has a 3D-geometry and rich semantic information enabling the automation of manual procedures. Because of this, BIM has the potential to enable a more objective data transfer and to eliminate hours of paperwork. That potential might be utilized for LCEA. But BIM is a recent technology, subject to ongoing development. Its large potential motivates to tailor solutions for the specific use case and to overcome reported shortcomings (Soust-Verdaguer et al. 2017).

1.2 Goal and Scope

To enhance the interoperability between BIM authoring software and LCEA applications, existing approaches for automated data transfer are researched. The researched approaches are tested employing exemplary software implementations and evaluated systematically. The findings serve as a basis to judge the most suitable approach that meets the introduced requirements of LCEA in the early design stages. This approach is customized and implemented in an own prototypical software solution, which is then coupled to a LCEA software to prove its functionality. Besides the functionality, the implementation is validated with regards to the algorithmics as well as the interaction with the user.

To delimit the scope of the work to a coverable extent, a focus is set to the German context of LCEA. This implies German datasets and standards as well as a moderate climate. Further, this thesis conducts applied research, to be industrially applicable in a short time horizon. Finally, to produce applicable results, the complexity of BIM-models has to be reduced to planar geometries (Rose und Bazjanac 2015) and the most important building element types.

1.3 Outline

In chapter 2, the theoretical background of BIM-based LCEA is introduced, divided into essentials of LCA, operational energy demand calculations, BIM, and the coupling of BIM and LCEA. The minimal semantic and geometric data required for the automation of the processes are worked out and lead to the method in chapter 3. The method begins with the research of existing approaches to interoperability. This is followed by a systematic evaluation of these approaches through a case study. Based on the findings, an enhanced data exchange procedure is designed and implemented. The results produced are critically reviewed before chapter 4 concludes with a summary and an outlook on potential developments in the field.

2 State of the Art

2.1 Life Cycle Assessment (LCA)

The International Standard defines LCA as “compilation and evaluation of the inputs, outputs, and the potential environmental impacts of a product system throughout its life cycle” (DIN EN ISO 14040). The scope of the life cycle is restricted by deliberately chosen system boundaries, the scope of impacts is restricted by the goal of the specific assessment. This section gives a brief overview of LCA.

2.1.1 Methodology according to DIN EN ISO 14040

An LCA according to the International Standard (DIN EN ISO 14040) is subdivided into four phases, as shown in Figure 2-1:

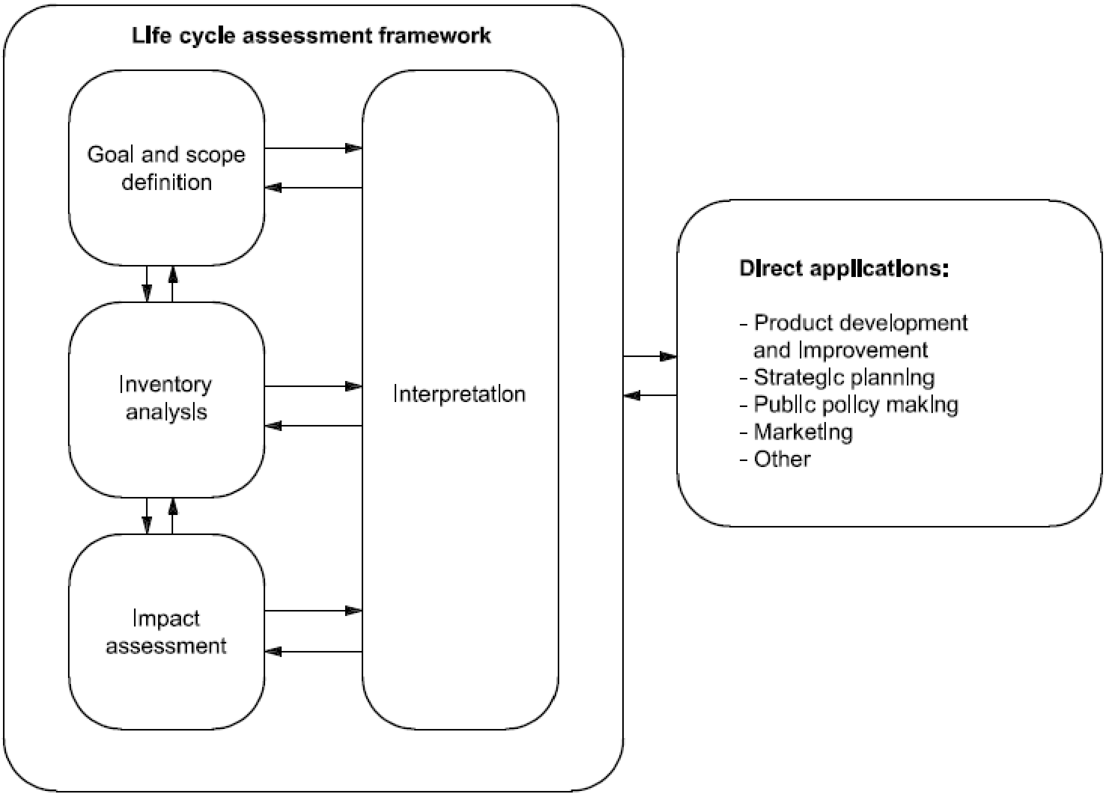


Figure 2-1: The four phases of LCA (DIN EN ISO 14040)

The first step is the definition of the goal. The goal is an intended gain of knowledge for a specific target group. This influences the scope and system boundaries to be considered, which in turn determine the effort to be applied for the life cycle inventory (LCI). The LCI is the information gathering that serves as a quantitative raw data basis for the following impact assessment (LCIA). The LCIA multiplies the raw data with impact factors from environmental databases to obtain impact indicators. These impact indicators quantify the environmental effects caused by the product system. Examples for these impact indicators are manifold. Table 2.1 gives an overview of the most important measures in Germany (König 2010), demonstrating the holism of LCA:

Table 2.1: Common Environmental Impacts used in LCA in Germany (König 2010)

Name	Abbreviation	Description
Ozone depletion potential	ODP	Potential contribution to the decrease of the ozone layer in the stratosphere. The ozone layer lowers the warming of the earth's surface and absorbs UV-radiation that is harmful to humans.
Acidification potential	AP	Contribution to the decrease of the pH-value of soil and water. The acidification causes damages to sensible ecosystems, e.g. acid rain to forests.
Eutrophication potential	EP	Eutrophication refers to the enrichment of nutrients at a certain location. Happening for example at bodies of water, algae growth may destabilize equilibriums.
Photochemical ozone creation potential	POCP	Potential contribution to the ozone increase in the troposphere. In comparison to the ozone in the stratosphere, the ozone near the earth's surface may cause direct damage to flora and fauna.
Global warming potential	GWP	Potential contribution to warming of the earth's atmosphere, caused by diffuse radiation of emitted gases.
Abiotic Depletion of Resources	ADP	Consumption of resources that become inaccessible for future generations

To conclude the LCA, the LCIA results have to be interpreted. In order to make the results comparable, the accumulated results are related to an illustrative and comprehensible functional unit. The interpretation phase includes a critical reflection of the methodology applied and thus may cause further iterations of LCA.

2.1.2 Life Cycle Energy Analysis

An input measure that plays a crucial role to obtain accurate impact indicators is the primary energy consumption (PEC). Thereby, especially the PEC through non-renewable sources is considered problematic (König 2010). It strongly influences the impact indicators in Table 2.1, e.g. the GWP and the ADP. To outline the considerations of PEC adequately, standards like the DIN EN 15804 distinguish the primary energy of non-renewable resources (PENRE) and material (PENRM) as well as the proportion of renewable resources (PERE) and materials (PERM). Making a balance of these proportions, the sum of the total non-renewable (PENRT) and the total renewable (PERT) primary energy consumption results in the PEC. Being in the design phase of a product system, consumption can not be measured. In this case, the estimated primary energy consumption is referred to as primary energy demand (PED).

As these granular considerations along the entire life cycle can become very complex, the field of life cycle energy analysis (LCEA) emerged. Ramesh et al. (2010) give an introduction to the method and describe its specialized implementation in architecture, engineering, and construction (AEC).

2.1.3 LCEA in Architecture, Engineering, and Construction (AEC)

2.1.3.1 Definition of the Life Cycle Phases

For LCA in AEC, specialized databases and standardized procedures have been developed. In the concerning European standard (DIN EN 15643-2), the life cycle of any product system “construction” is structured as shown in Figure 2-2:

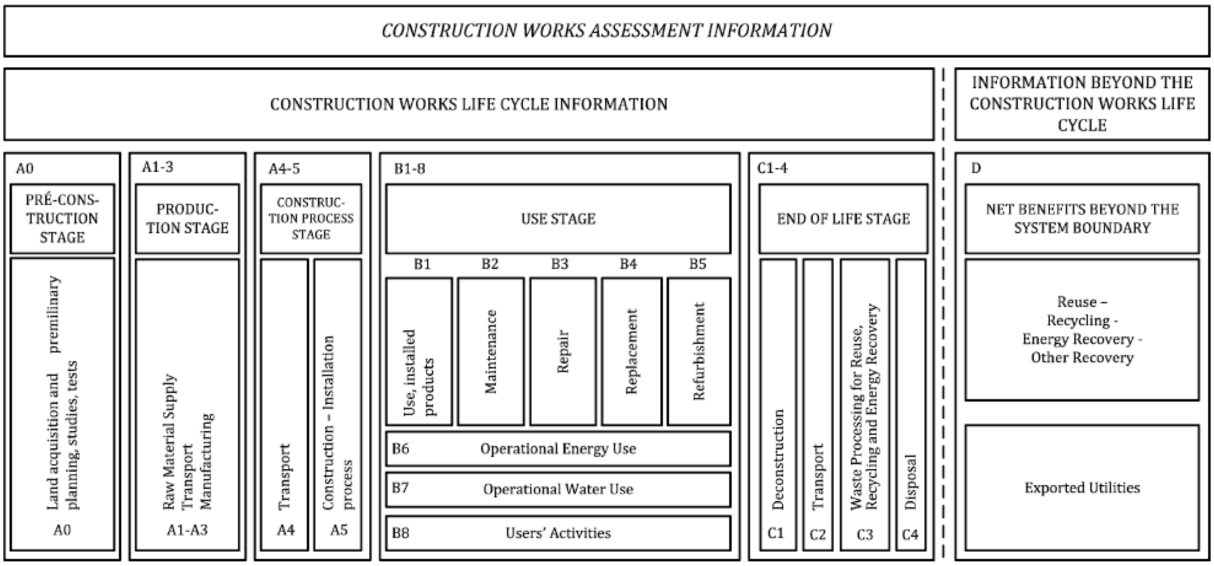


Figure 2-2: Life cycle division in AEC according to DIN EN 15643-2

The modules A0-A5 include the processes before the use stage of the construction, including the construction work (A4/A5). Modules B1-B5 include the embodied impacts caused during the use stage, e.g. through retrofitting or repairs. B6 to B8 refer to impacts caused by the operation of the building, e.g. for heating or water consumption. modules C1-C4 include the deconstruction and waste disposal of the materials, and module D considers energetical savings outside the system boundary (Cooper 2014).

When analyzing the life cycle energy of a building, a distinction between embodied energy (EE) of materials assembled and the operational energy (OE) caused by the building's use is necessary. Both measures require different input information and calculation procedures (Ramesh et al. 2010). Due to each field's complexity, software and research on energy efficiency mostly focus on one component or the other (Dixit 2017). Statal regulations in the past put an exclusive emphasis on OE. Therefore, EE considerations become more relevant for more recent buildings, as shown in Figure 2-3:

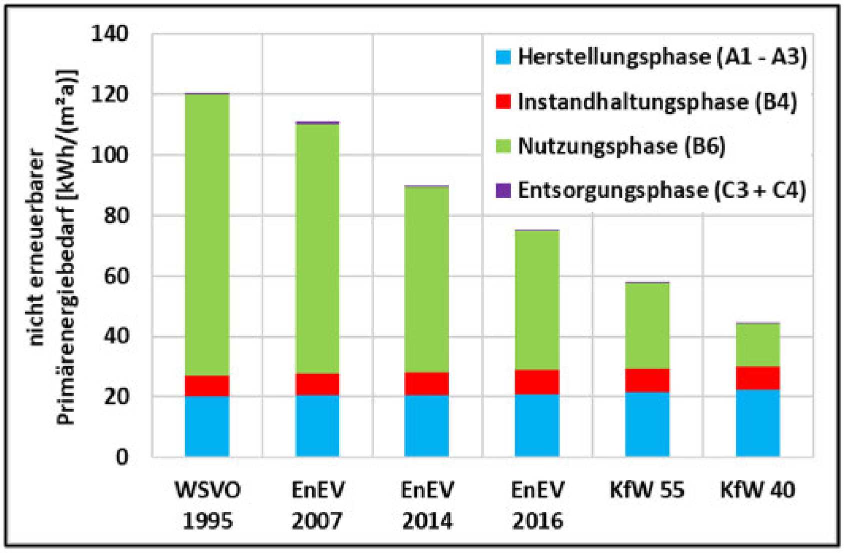


Figure 2-3: Development of OE and EE ratios on PED for multiple-family houses (Schöndube 2016)

In the figure, the horizontal axis shows the requirements of German regulations on OE from the past (left) to the present (right). The green part of the bars indicates the extent of OE demand, the others accumulate to EE. The increase in the proportion of EE and the relevance of a combined consideration of EE and OE is evident.

2.1.3.2 Embodied Energy Considerations

For EE considerations, the LCI requires a generation of a bill of material quantities (BoMQ) according to the scope set. To conduct the LCIA and to match the BoMQ to impact indicators, established databases on common construction materials exist, e.g. the ÖKOBAUDAT (Bundesregierung 2019) in Germany. They provide data based on environmental product declarations (EPD) of vendors as well as generic, industry-wide data. Given a BoMQ, the LCIA then is a simple sum of multiplications of the materials' masses (m) with the materials' primary energy demands over the life cycle phases, see equation (1) (Harter et al. 2020).

$$\mathrm{EE} = \mathrm{m} \times \sum_{i=A1}^{C4} PEi \qquad (1)$$

The environmental databases commonly do not provide data on every single life cycle category according to Figure 2-2. It is difficult to statistically predict the construction phase (A4-A5), the long use phase (B1-B8), or recycling and disposal (C2-C4) in the far future (König 2010). To remedy this limitation, one possibility is to reduce the scope of the LCA accordingly. If a reduction of scope makes it impossible to meet the goal of the LCA, project-specific considerations can be made. This is the case for the OE demand during LCEA in AEC.

2.1.3.3 Operational Energy Considerations

Although it gives only information about one module of the life cycle of a building (B6), OE demand calculations are complex and require a multitude of data. The input for OE demand calculation software ranges from information on heating, ventilation, and air conditioning (HVAC) systems up to usage characteristics of the building. While EE demand in Germany plays a role only in the relatively small niche of sustainability certifications (Forth 2018), OE demand is and has been subject to complex regulations (Bundesregierung 2013). Those regulations play a crucial role in every planning process. Because of this, OE demand calculations are introduced, referring to common regulations in Germany.

2.2 Operational Energy Demand Calculation for Buildings

In 2.2.1, the calculation of the useful heat demand is introduced, which serves as a basis for the PED calculation in 2.2.2. The procedures follow the industrial standard DIN 18599 that is commonly applied in the context of legislation. Therefore, the concerning German regulation is discussed in 2.2.3.

2.2.1 Useful Heat Demand Calculation according to DIN 18599

In Germany, the standard DIN EN 18599 is the most influential set of rules referred to for OE calculations, even if less recent standards like the DIN 4108 are legally applicable as well.

To calculate the PED according to DIN 18599, the useful heat demand of a building is the basis. This is calculated as the difference between manifold heat sources and heat sinks. In construction physics, transmissive, radiative, and ventilative sinks and sources are distinguished. For moderate climates, only five types of measures have to be calculated during OE demand calculation. These are listed in Table 2.2, with their minimal data requirements to be retrieved from a practitioner:

Table 2.2: Energy Measures for OE demand calculations according to DIN 18599

Type	Description	Name (DIN 18599)	Minimal Geometric Data
Transmissive Losses	Transmissive heat losses from diffusion between warmer thermal zones in a building and the outside atmosphere.	$Q_{T,Sink}$	Envelope Boundary surfaces by zone incl. thermal properties of building elements
Solar Gains	Radiative entries of sun energy into the building envelope, hereby transparent and opaque building elements, their orientation and shading have to be distinguished	$Q_{sol,trans}$ $Q_{source, op}$	Transparent and Opaque Element Geometry of Building Envelope grouped to zones
Ventilation Heat Losses	Heat losses transferred via air flow and openings.	$Q_{sink,V}$	Gross Volume Ve of building envelope
Internal Loads	Heat entries through men and MEP inside a building envelope	Q_I	Floor Area and storey height grouped to zone
Useful Drinking Warm Water Heat Demand	Amount of heat that is consumed through the drinking warm water, depending on the usage characteristics of a building.	$Q_{w,b}$	Floor Area grouped to zone

This textual description may be illustrated by Figure 2-5, the single measures are discussed in the following sections.

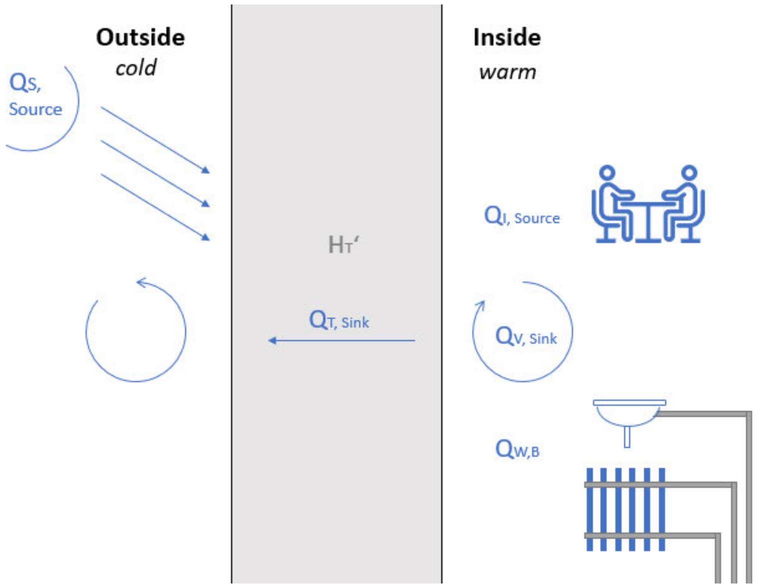


Figure 2-4: Relevant heat sinks and sources in moderate climates (own compilation)

2.2.1.1 Transmissive Heat Losses

The calculation of the transmissive measures is the most time-consuming and complicated part for the user of OE demand calculation software (Lichtmeß 2010), as it requires thermal zoning of the building and a measurement of the building envelope according to thermal zones. A thermal zone is a group of rooms in a building with homogenous usage characteristics and thermal boundary conditions, maintained by HVAC systems (Kempf 2011). According to the assigned zone, usage characteristics can be retrieved from standards, e.g. the warm water consumption per m² for a toilet. Kempf (2011) gives an example of a zoned and consistently colored floor plan:

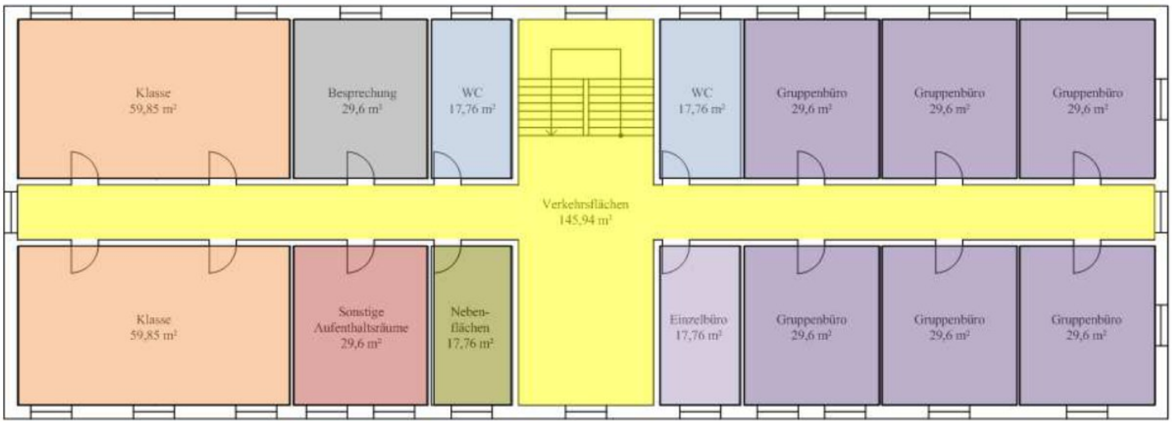


Figure 2-5: Example of a zoned building (Kempf 2011)

For every zone, a heat loss coefficient H_T' [W/K] may be calculated according to equation (2), that then in simulations or steady-state methods is integrated with time and climate data to the transmissive heat loss $Q_{T,\, Sink}$.

$$H_T' = \sum_{i=1}^{n}(Fxi * Ui * Ai) + x * A \qquad (2)$$

F_x [-] is a correction factor considering the thermal boundary condition of a building element, depending on its' function, inclination, and contact medium. U [W/(K*m²)] is the heat transmission coefficient and x [W/(K*m²)] is a coefficient to consider globally heat bridges and A [m²] refers to the area of every element of the building envelope.

2.2.1.2 Other Factors influencing the useful Heat Demand

Solar heat entries are calculated according to equation (3)

$$Q_S = \sum_{i=1}^{n} Is, i * \ (0{,}567 * gi * Ai) \qquad (3)$$

Whereby Is [W/m²] is the solar intensity depending on the orientation of the building element and the climate and season. The coefficient g [-] indicates the transmissivity of glass and A [m²] the surface of the windows.

Again based on the zoning, internal loads and the warm water heat demand can be calculated. Reference data from standards like the DIN 18599 can be used. Alternatively, a proficient user can also specify occupancy schedules and consumption rates.

To calculate ventilation heat losses, the net volume and the air exchange rate per zone have to be known. Further, the airtightness of the zone boundaries plays a role.

Given the discussed five measures, the PED may be deducted from the resulting gap of the sinks less the sources, the useful heat demand.

2.2.2 PED calculation

To calculate the PED, the useful heat demand has to be generated by the mechanical equipment (MEP) of the building. The efficiency of the MEP is considered by backpropagation through the distribution system(s) and the devices they lead into. This results in the final energy demand of a building that is generated in-place. The final energy demand [W] is multiplied with a primary energy factor considering the source of energy to obtain the PED [W], including the entire supply chain of the energy sources. At this point, the distinction between non-renewable PENRT and renewable (PERT) sources of PED must be applied. For the environmental impact, the non-renewable proportion is crucial, as PENRT influences stronger the introduced impact indicators, e.g. the GWP or the ADP (König 2010).

The DIN 18599 is merely an industrial standard and is usually applied as a reference of legislation. The German Energieeinsparverordnung (EnEV) refers to the DIN 18599 for its calculation procedures.

2.2.3 The German Energy Saving Ordinance EnEV

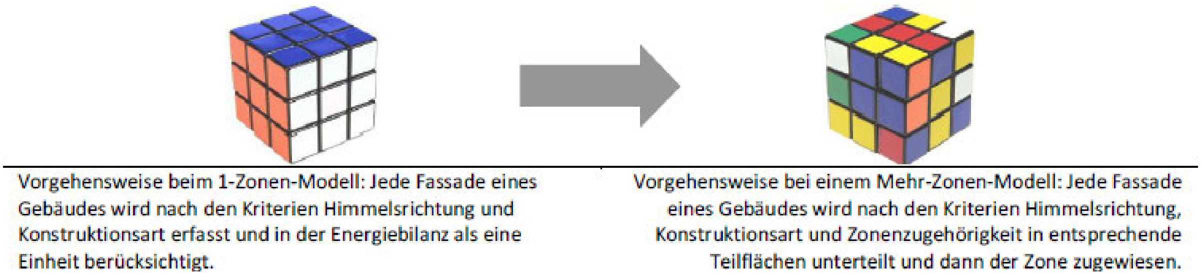
The German EnEV defines the energetic requirements of buildings, distinguished to the use cases renovation and newly designed buildings and the building types residential and non-residential buildings.

2.2.3.1 Comparative Approach of EnEV

OE demand calculation is not a way of predicting real PEC, but a qualitative, comparative assessment instrument. Following this logic, the EnEV introduced a method where the energy performance of any building in design is compared to a so-called reference building. The reference building is a virtual base of comparison with equal geometry and default values for U-values, airtightness, HVAC efficiency, and other measures. Depending on the use case and the building type, comparatively good performance has to be proven, according to the reference model has to be proven. Of course, the PED has to undercut the performance of the reference building to a certain extent. For newly designed residential buildings, the measure H_T‘ has to be less than a certain percentage of the reference building to ensure high insulation. Also, U-values being less than the reference values certain building elements must be proven, e.g. for external walls. Finally, it is noteworthy that the EnEV has to be applied in combination with the law on renewable energy GEG (Bundesregierung 2020) to reduce the non-renewable ratio of PED.

2.2.3.2 Zoning and Calculation Methods according to EnEV

The extraction of the building envelope surfaces according to zones is the most time-consuming step in an OE demand calculation (Lichtmeß 2010). Therefore, the EnEV and many scientific publications seek to enable an omission or postponement of the zoning (Hauser 2006; Lichtmeß 2010). In the case of omission, is commonly referred to as the “one zone model” approach. The left compilation in Figure 2-6 shows the simplification achieved. However, it is important to highlight that the term “one zone model” might be misleading, as the zoning (right in Figure 2-6) is then introduced in the post-processing of OE demand calculation software.



Vorgehensweise beim 1-Zonen-Modell: Jede Fassade eines Gebäudes wird nach den Kriterien Himmelsrichtung und Konstruktionsart erfasst und in der Energiebilanz als eine Einheit berücksichtigt.

Vorgehensweise bei einem Mehr-Zonen-Modell: Jede Fassade eines Gebäudes wird nach den Kriterien Himmelsrichtung, Konstruktionsart und Zonenzugehörigkeit in entsprechende Teilflächen unterteilt und dann der Zone zugewiesen.

Figure 2-6: Simplified retrieval of building envelope without distinction of zones (left) and with the distinction of zones (right) (Lichtmeß 2010)

The EnEV does not prescribe a multi-zone envelope quantity take-off (QTO) for all use cases and building types. For residential buildings, the “one zone model” approach is always allowed, given that the building is not cooled. For the ministry issuing the EnEV, Hauser (2006) developed a procedure to apply the one zone QTO also to non-residential buildings.

Having introduced EE and OE calculation procedures, the early design stage is deepened in its special requirements and challenges.

2.3 LCEA in AEC in Early Design Stages

2.3.1 Motivation

In the industrial practice of LCEA, the methodology is used with a very different goal and scope in the early and in the late design stages (Hollberg 2016). Therefore, it is necessary to consider both two distinct use cases of LCEA.

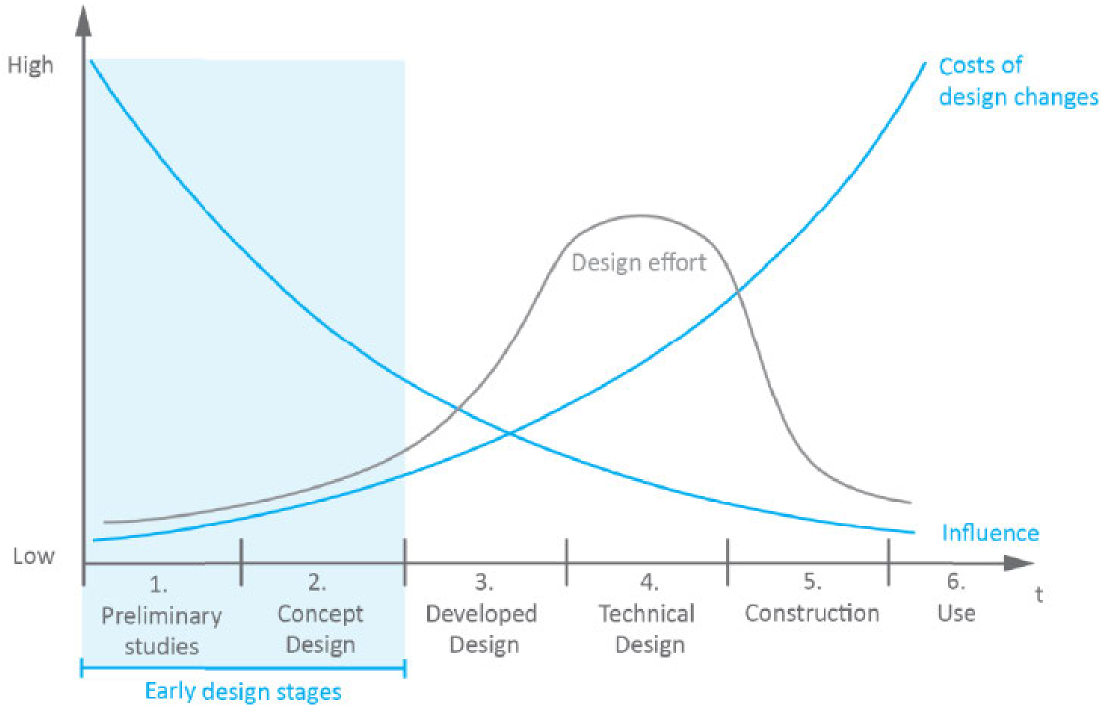


Figure 2-7: Cost of design changes according to design progress (Paulson 1976)

The Paulson-Curve, Figure 2-7, shows that the early design stage offers great potential in optimizing the life cycle energy performance. Design changes require little effort and have a high influence. Unfortunately, in the early design stages, most of the data is unknown, posing the challenge to cope with this uncertainty.

On the contrary, late design stages are characterized by low flexibility and high efforts for design decisions. Because of this, the goal of an LCEA in this stage is mainly the abidance of regulations and certification standards (Hollberg et al. 2017). Due to the progress in design, the scope of the LCEA can be wider and the LCI is more reliable.

The great potentials in energy efficiency encourage to cope with the challenges LCEA poses in early design stages. It was demonstrated that the environmental impact of a benchmarked building could then be reduced by 40% (European Commission 2008). In the following, basic concepts for remedying the challenge of low data availability are introduced.

2.3.2 Limitations in Data Availability in early Design Stages

2.3.2.1 Screening LCA

The International Reference Life Cycle Data (ILCD)-handbook (European Commission 2011) deals with the problem of low data availability. It points out that LCA can be conducted iteratively, with an increasing degree of detail, see Figure 2-8:

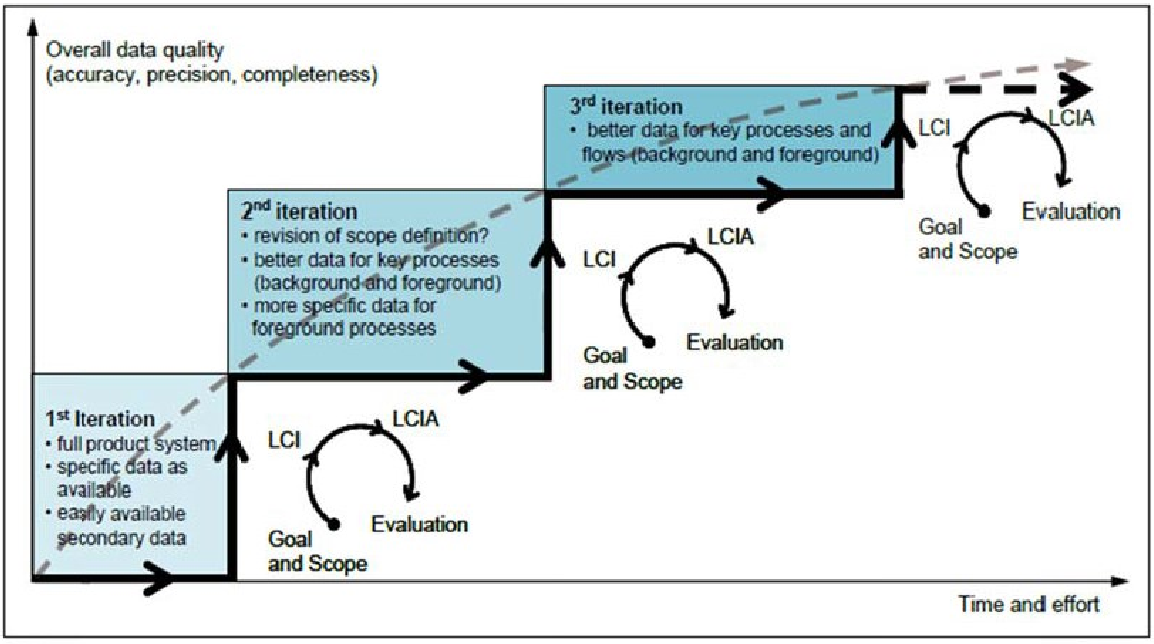


Figure 2-8: Concept to increase the level of detail of LCA (JRC European Commission 2011)

Consistently, the operational guide to the ILCD-handbook (European Commission 2011) proposes two simplifications in comparison to the ‘Complete LCA’: The ‘screening LCA’ and the ‘simplified LCA’. The screening LCA thereby is the vaguest prediction and reduces the data requirements the most.

2.3.2.2 Screening LCA for AEC

An example of such a ‘screening LCA’ for AEC describes Hollberg (2016), including a minimalist implementation of OE demand calculations, described in Hollberg et al. (2017).

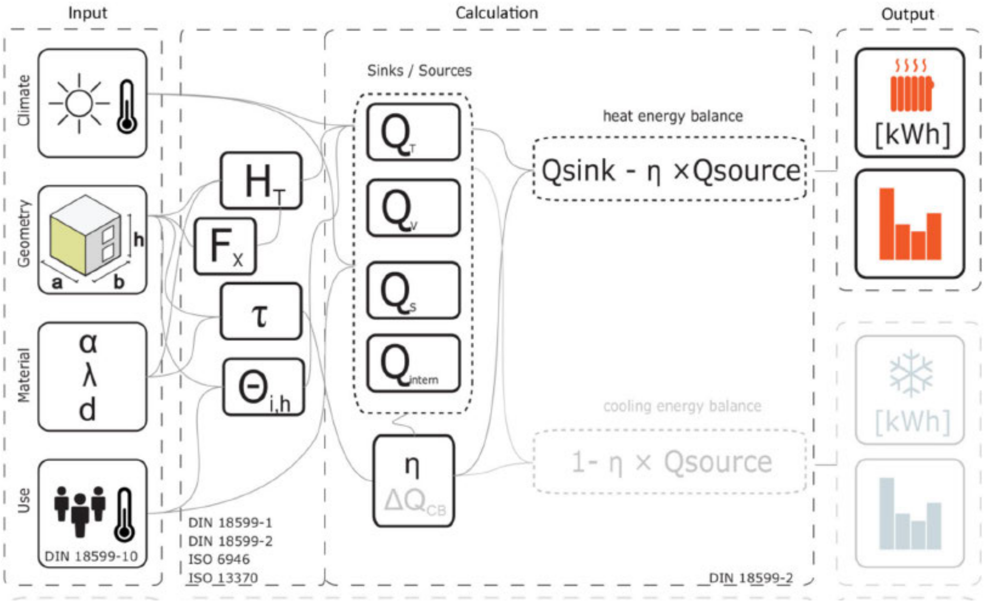


Figure 2-9: Visualization of simplified heat warmth demand calculation (Hollberg et al. 2017)

The approach drastically reduces the data requirements compared to established OE demand calculation software.

- Geometrical data requirements are reduced as only the envelope is extracted without distinction of zones
- Several possible heat sources and heat sinks according to DIN 18599 are neglected, e.g. cooling or lighting. The researcher follows the minimum required by the EnEV (Bundesregierung 2013).
- The fastest and simplest, but also most imprecise monthly discretization is applied (the monthly steady-state procedure).
- The materials of building elements are assumed to be uncertain and therefore used as a variable of input

The latter simplification may be better understood by considering Figure 2-10 that shows the different levels of granularity a building can be analyzed:

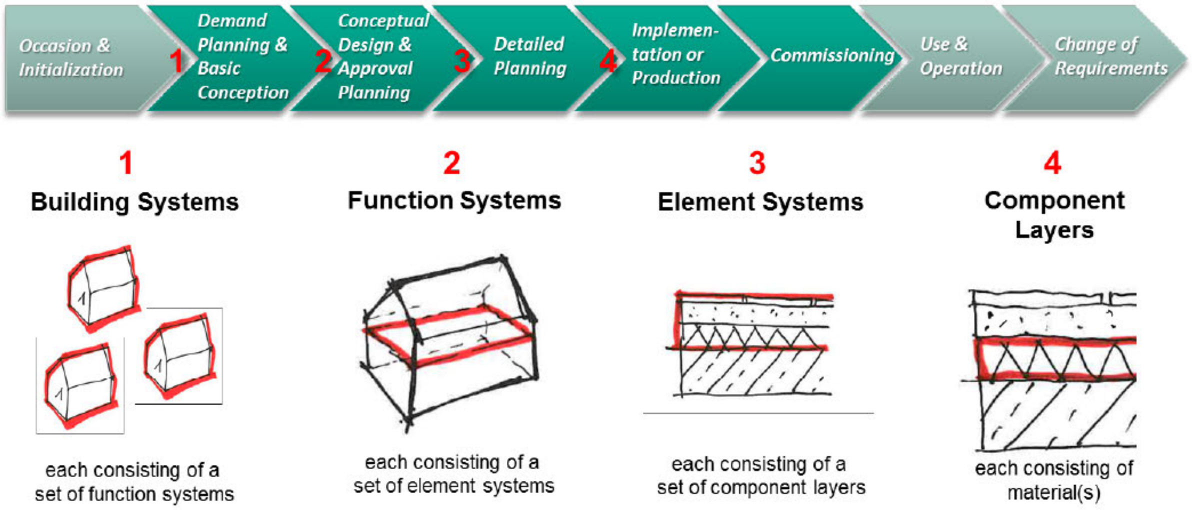


Figure 2-10: Detail levels considerable in LCA method (Ebertshäuser et al. 2019)

Hollberg (2016) proposes an operation on the “Function Systems” level for LCEA in the early design stages. This means that constructions are assigned variably to building elements, from a building component catalog. Like that, a fast generation of variants with very different material properties is possible. Most OE demand calculation software work on an “Element Systems” level and therefore require higher effort to compare different methods of building.

To conclude the introduction to screening LCA, the ILCD handbook recommends a justification of simplifications applied and a critical review of its significance.

2.3.3 Critical Review of screening LCEA Method

Any simplification made in a screening LCA must be compared and justified in comparison to the full LCA with established methodology (JRC European Commission 2011). However, it must be taken into consideration that full LCA in AEC commonly pursue another goal. While full LCAs using complex OE calculation software are intended for scientific purposes or to meet regulations, the screening LCEA in early design stages shall serve as a didactic design-support. Therefore, it is important that results obtained still indicate sensitivities in the current state of design, but do not have to be quantitatively precise.

For EE, it is important to differentiate errors caused by outstanding design decisions and uncertainties arising from the impact factors. König (2010) found the deviation of EE impact factors from environmental datasets like the ÖKOBAUDAT to be negligible. It is rather important to communicate the design uncertainty to the user. To communicate design uncertainty, several approaches exist (Schneider-Marin et al.

2020; Ebertshäuser et al. 2019). Thereby, this uncertainty should be considered potential and guidance in the design process. For OE, the statements on design uncertainty are equally valid, but at least two other factors must be considered additionally:

First, Hauser (2006) or Keltsch (2019) quantified the inaccuracy caused by the omission of precise thermal zoning. Hauser (2006) recommends an omission of precise zoning only if certain conditions are met. For buildings with automated ventilation or great differences in the thermal conditioning, the deviations become too high to guide optimization reasonably. Therefore, it is important to instruct users of LCEA on the conditions that allow an omission of thermal zoning. Second, Hollberg et al. (2017) describe the possible inaccuracy through the choice of the fast monthly steady-state calculation procedure in his method. Again, it is important to consider the assumptions made to develop the approach. The monthly steady-state approach may become more inaccurate in microclimatic situations where diurnal temperature fluctuations are higher, as it can only approximate this data. This discussion on simplifications made could be continued, pointing to the same conclusion: If the user is communicated the simplifications made and the conditions they are valid on, a screening LCEA may serve as a frequently accessible, didactic design support.

But not only the described methodological simplifications are sufficient for this purpose, also technological support is necessary. To achieve a meaningful and fast data exchange, building information modeling (BIM) shall be used for the extraction of the building geometry and required semantic information.

2.4 BIM

2.4.1 Definition

A building information model is a digital depiction of a construction with a great depth of information (Borrmann et al. 2018). Besides the 3D-geometry, also semantic information may be stored. Building Information Modeling then describes the processes of creating, manipulating, and administrating the BIM-Model using Software-tools (Borrmann et al. 2018). Further, BIM includes the model’s use for other software-applications throughout the whole life cycle of the construction.

In a construction project, many stakeholders from different fields communicate and exchange information, e.g. the architect and the construction firm. At every interface, there is a risk of losing information. Figure 2-11 visualizes this:

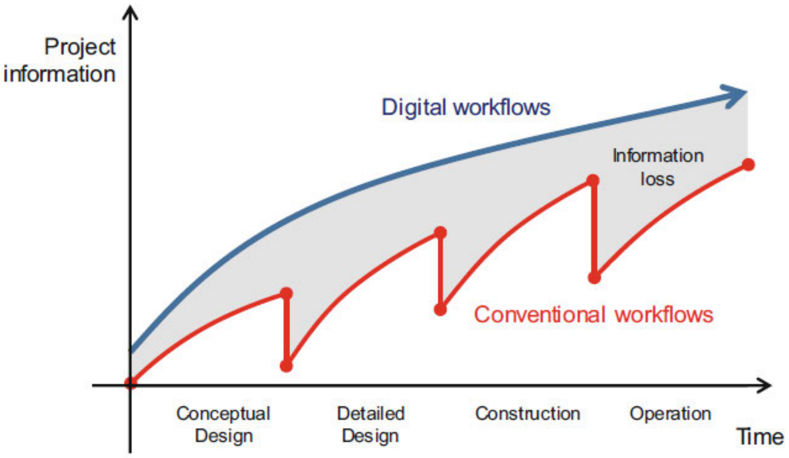


Figure 2-11: Data continuity in conventional and in digital workflows (Sacks et al. 2018)

In contrast to other technical industries, the construction industry reuses previously generated digital data to a smaller extent (Borrmann et al. 2018). Therefore, the aim of BIM is the continuous re-use of data from the centralized BIM-model. This should result in a higher consistency as well as a higher efficiency of processes. Bazjanac et al. (2011) confirm that utilizing BIM, OE demand calculation software can become more objective.

2.4.2 BIM-Modeling

2.4.2.1 Geometric Modeling

The 3D geometrical representation of a construction is the base of every BIM-Model. Thereby, single building elements are described as 3D-solids with two main approaches to store them: The implicit and the explicit description. Implicit geometry descriptions define a volume by a sequence of algebraic and geometric procedures, e.g. extrusions, rotations, intersections, or similar. Repeating those steps, a CAD engine may generate the same model. Constructive Solid Geometry (CSG) is the most known implementation of implicit representations (Sacks et al. 2018).

Explicit representations do not contain the construction history, but merely the resulting geometry and the belonging topology. The Boundary Representation (B-Rep) hereby is the most common implementation (Borrmann et al. 2018). In a B-Rep, the shape is stored as a closed, oriented set of surfaces. Hierarchically, the (sur-)faces are a composition of edges and the edges a composition of vertices. Topology and geometry are stored separately, which makes B-Rep flexible, as the geometry linked to the topology can be any numerically implemented free-form-curve or free-form-surface. An example of a B-Rep pyramid is displayed in Figure 2-12:

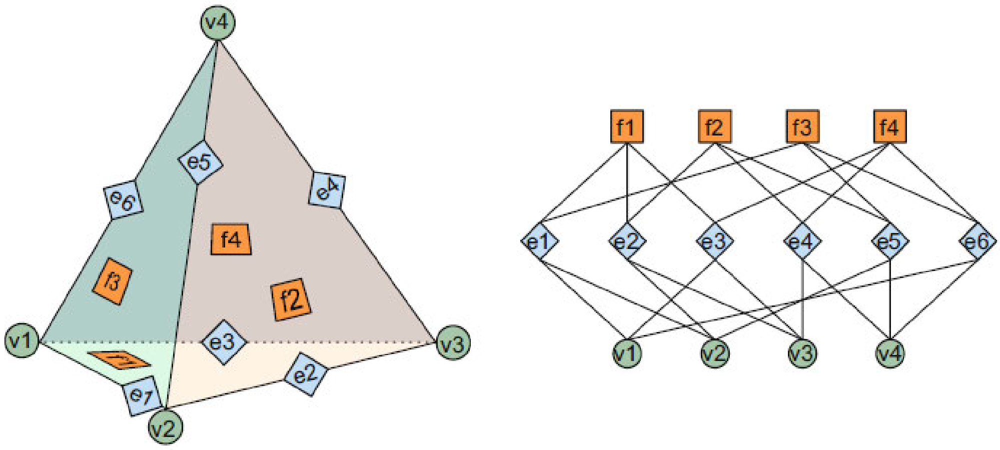


Figure 2-12: Storage of a simple B-Rep with the vef-notation (Borrmann et al. 2018)

While at the beginning of computer aided design (CAD) implicit and explicit descriptions were “competing for supremacy” (Sacks et al. 2018), today’s BIM-authoring software uses them in parallel. However, to downstream applications, commonly explicit geometries are transferred. As they store the resulting geometric data, errors in the replication of implicit procedures may be avoided.

2.4.2.2 Semantic and Parametric Modeling

A BIM-Model, besides the geometry, also contains semantic information. This results in a great depth of information a BIM-Model contains. To cope with this, the encapsulation of information and functionality is necessary. Therefore, the paradigm of object-oriented programming is applied, with each object storing attributes and methods. To illustrate the object-oriented principle, any wall object contains a variety of properties, to describe its material, fire safety, or thermal characteristics. The methods of a wall class then enable a person to access and modify them efficiently.

In BIM-modeling, the objects’ geometric and semantic attributes are not treated as fixed values, but as flexible parameters (Borrmann et al. 2018). Thereby, it is important to highlight that authoring software implement parametric modeling with a gradual degree of flexibility (Borrmann et al. 2018). Some parameters might be adapted on an object instance level, while others must be determined in the object templates

(“Families” in Revit). Giving an example, a user may change the orientation of a window for every instance, while the clearance width can only be modified in the window template. In the case of the latter, all instantiations of the modified window template would be changed at once.

The concept of parameters is applied in the method part of the thesis. There, parameters are used to calculate the building element’s geometry and to enable robust interoperability between BIM-authoring tools and a LCEA software.

2.4.3 Interoperability and Exchange Formats

2.4.3.1 Interoperability

The adoption of BIM varies largely between different countries and different companies. When the client commissions the planners and construction firms, the usage of BIM requires the interoperability of the individual software solutions every player uses. Thereby, a distinction between “open” and “closed” BIM is made, which refers to the formats used for the exchange of data. Closed BIM uses commercial formats like RVT (Autodesk Revit 2019) or PLN (Nemetschek Graphisoft) that are proprietary, while open-BIM approaches rely on vendor-neutral, open-source formats like the Industry Foundation Class (IFC) or Green Building XML(gbXML). Those exchange formats are developed by non-profit organizations to be independent of specific vendors and bear the potential to lower the number of interfaces to be developed and hence the quality and cost of development for BIM-based applications. But the usage of open exchange formats also introduces an additional interface to trespass, bearing a potential loss of information and a complication of the workflow compared to a solution embedded in the authoring format. These are challenges to be accounted for when designing a data exchange between two BIM-compatible applications.

2.4.3.2 Exchange Formats

Malhotra et al. (2019) conducted a literature review on interoperability for energy analysis and evaluated the three open formats IFC, gbXML, and CityGML to couple the method to a BIM-model. CityGML is focused on the macroscopic analysis of the built environment, what is not the focus of this thesis, and therefore neglected. IFC is a multi-purpose data exchange standard developed and maintained by buildingSMART International (2020). The standard arrived at a fourth version with

IFCx4 in 2016 and departed as a specification of the standard for the exchange of product model data (STEP) (ISO 10303). IFC is a complex and extensive data format as it pursues to enable the data exchange for a wide range of software applications in AEC. When using IFC for a downstream application, exporting to the schema to meet the diverse needs of the customized processing chain is a demanding task.

GbXML on the contrary aims to cover only the use case of energy considerations, The format is developed by the US American company Building Green Inc (2020). Since 2009 it is a stand-alone entity maintained by the non-profit organization “Open Green Building XML” (Green Building Foundation 2020). As implied in its name, it can store a building’s environmental information in an easily readable extension markup language (XML) schema. The BIM model reduced to the environmental aspects is often referred to as the building energy model (BEM). The schema of gbXML is divided into a header and many subcategories per project, for LCEA in early design stages at least the following elements would be necessary:

- One Campus as a container for building(s)
- Building(s) with an assembly of a closed building envelope as a container for surface(s)
- Surface(s) with surface type specification (e.g. “InteriorWall”), B-Rep geometry, references to adjacent spaces, and assigned openings
- Spaces referencing to zones

IFC and gbXML are exchange formats that may be interpreted by LCEA software. In the following, the use of the data from a BIM model is contextualized.

2.5 BIM-based LCEA in Early Design Stages

Figure 2-13 visualizes the procedure of LCEA based upon a BIM model (Basbagill et al. 2013; Hollberg 2016):

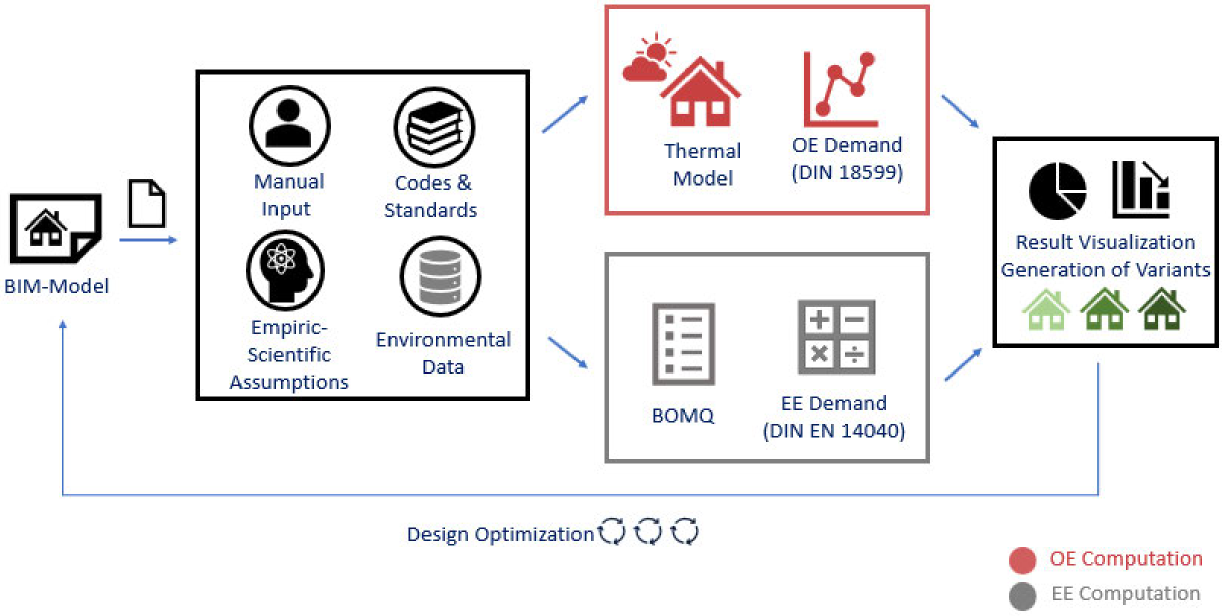


Figure 2-13: Process map of BIM-based LCEA (own compilation)

In LCEA software, the data parsed from a BIM format is enriched with a variety of data. For example, the user specifies the building type, the microclimatic situation, confirms the materials used, and selects MEP configurations. In section 2.1.3.2 and section 2.2, the overall data requirements and procedures to calculate EE and OE were discussed. Therefore, this section focuses on the data that must be retrieved from BIM-models. According to the use case, the interface implemented must consider the high uncertainty in the early design stages. As discussed in 2.3.2.1, most semantic data should then be considered variable and be treated as parameters to optimize the energetic performance of the building. Given this, the main purpose of BIM-models is a fast and consistent transfer of geometric data, including minimalist energetic information. In the following, the BIM-integration EE and OE calculations are introduced, subsequently commercial solutions in the field of LCEA in early design stages.

2.5.1 BIM integration for EE calculations in LCEA in Early Design Stages

The basis for EE demand calculation in LCEA is a BoMQ from an architectural BIM-model. The BOQ is structured into groups of building elements according to their type, functionality, and/or materials to have an outline for the LCEA results and to enable a possible optimization. During the export, a mapping of the building elements' materials to the impact indicator databases of the LCA application has to be established. Forth (2018) conducted a case study and found this step to be error-prone. Therefore, he recommends a semi-automated procedure. As shown in Figure 2-14, between LCI ("Sachbilanz") and LCIA ("Wirkungsabschätzung"), manual confirmation of the correct linking to environmental data is introduced.

Figure 2-14: Procedure for semi-automated, BIM-based EE calculation (Forth 2018)

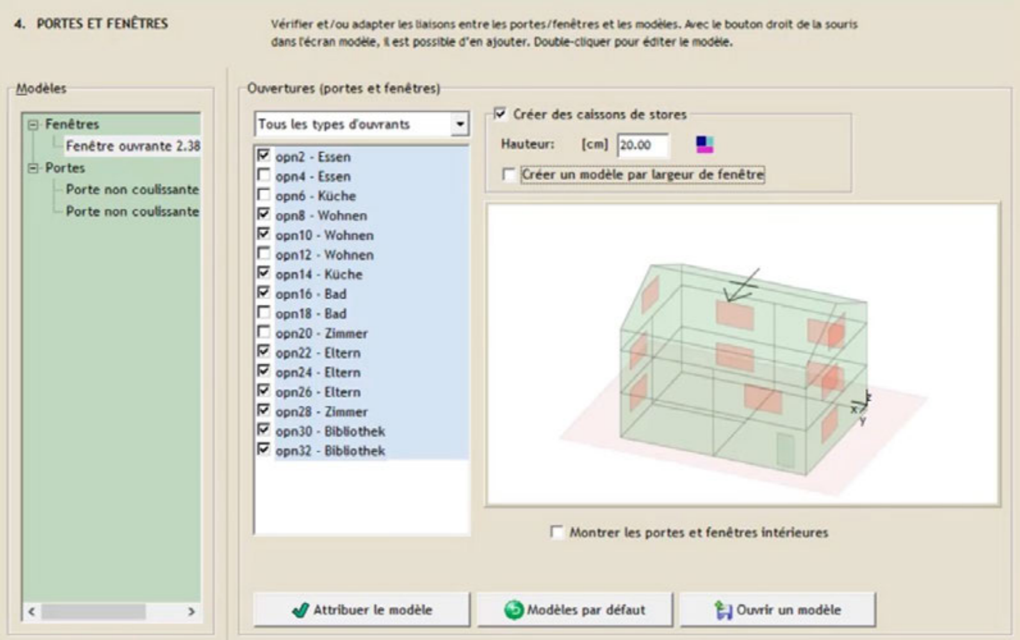
For early design stages, mapping and optimization of materials commonly take place on the "functional systems" level (Ebertshäuser et al. 2019). This level considers the building elements' properties variable. Referring to building component catalog instead of modifying single materials saves time and is consistent with the high data uncertainty in the early design stages. Information on the layers of a building element is likely to be unknown and requiring this from a user would discourage him to apply the methodology.

2.5.2 BIM integration for OE Demand Calculations in early Design Stages

To calculate the OE demand, at least the data described in Table 2.2 has to be retrieved, see section 2.2.1. To conduct thermal zoning, which is a requirement for any of the listed measures, in BIM-software 3D-space objects are placed - "MEP-rooms" in Revit, "Zones" in ArchiCAD. These objects yield functionality to define the building envelope surfaces distinct to zones, to calculate the zone areas and gross volume of the building. The information of the space-objects then may be transferred to OE demand calculation software by any format introduced.

A conversion from BIM to BEM conversion is a complex task and subject to ongoing research. The International Building Performance Simulation Association (IBPSA), authoring the software EnergyPlus, in May 2020 held a seminar with the title "BIM to BEM: Mission Impossible or Pursuit of Hapiness?" (IBPSA 2020). This points out that in industrial practice there are still severe limitations in the interoperability from BIM to OE demand calculation software. Especially the geometric processing is error-prone

(O'Donnell et al. 2013; Rose und Bazjanac 2015). Because of these difficulties, a visual feedback for a BEM model checking is crucial. For the gbXML export, authoring tools (Revit, ArchiCAD) have implemented functionality to indicate roughly the analytical surfaces to be exported. For a more detailed checking including surface properties, XML-viewers like the FZK-viewer or Aragog of Ladybug may display an exported gbXML. Finally, when importing the gbXML into the LCEA software, some software vendors offer a visual import wizard, e.g. Ecosai Lesosai (2020), as shown in Figure 2-15. During the import, minor semantic corrections may be done.



4. PORTES ET FENÊTRES

Vérifier et/ou adapter les liaisons entre les portes/fenêtres et les modèles. Avec le bouton droit de la souris dans l'écran modèle, il est possible d'en ajouter. Double-cliquer pour éditer le modèle.

Modèles
- Fenêtres
  - Fenêtre ouvrante 2.38
- Portes
  - Porte non coulissante
  - Porte non coulissante

Ouvertures (portes et fenêtres)

Tous les types d'ouvrants

☑ opn2 - Essen
☐ opn4 - Essen
☐ opn6 - Küche
☑ opn8 - Wohnen
☑ opn10 - Wohnen
☐ opn12 - Wohnen
☑ opn14 - Küche
☑ opn16 - Bad
☐ opn18 - Bad
☐ opn20 - Zimmer
☑ opn22 - Eltern
☑ opn24 - Eltern
☑ opn26 - Eltern
☑ opn28 - Zimmer
☑ opn30 - Bibliothek
☑ opn32 - Bibliothek

☑ Créer des caissons de stores

Hauteur: [cm] 20.00

☐ Créer un modèle par largeur de fenêtre

☐ Montrer les portes et fenêtres intérieures

Figure 2-15: GbXML-import wizard inside the application (Ecosai Lesosai 2020)

BIM-based LCEA implies the interaction between authoring tools and the downstream-applications retrieving the data from the BIM-model. Exemplary, some commercial solutions are presented to contextualize the work on interoperability.

2.5.3 Commercial Solutions for BIM-based LCEA in early Design Stages

For the research of commercial solutions, two criteria were applied: First, the possibility for a user to retrieve a combined OE and EE feedback. This means a closed-loop of computation and a combined result visualization. Second, the effort of the software usage that should encourage users to frequent and didactic use of LCEA.

Regarding the first criterion, currently, OE and EE calculations are rarely implemented together. Some OE demand calculation software offer built-in functionalities to calculate EE demand as well because the EE ratios matter for sustainability certifications like BREEAM or DGNB. Exemplarily, the swiss software Ecosai Lesosai (2020) is referred to. EE calculation software, e.g.Tally, similarly offers the possibility to insert a value for the OE demand. Others implemented an interface with an OE demand calculation software, e.g. oneClickLCA with DesignBuilder. Regarding the second criterion, especially OE demand calculation software overdemands the users in early design stages. The software is designed to be understood by energy consultants, but the person responsible for the design is rather an architect (Schlueter und Thesseling 2009). Most data requirements are expected to be fixed but are rather variable (König 2010). The result visualization must be design-guiding but is rather regulation-conform or scientific (Hollberg 2016).

In Germany, the only commercial solution found to meet the two criteria described is CAALA (2020). The company implements to a great extent the scientific approach of Hollberg (2016), introduced in section 2.3.2. Hollberg (2016) simplified the data requirements to address users that are not proficient in energy considerations. As well, CAALA (2020) accounted for the goal of LCEA to be a design-guiding method: On one hand, it considers most input parameters variable parameters. On the other hand, it offers a result display with few charts that aims to depict the impact of design decisions through few numerical values. To encourage users to frequent usage, it enables real-time feedback when using connected sketching software. However, a limitation of CAALA (2020) is certainly the current BIM-integration. There is no real-time functionality, only a gbXML-import. This gbXML-interface does not include a visualization like Ecosai Lesosai (2020) implemented it. Further, BIM modeling guidelines are missing as well as instructions on the export of the format. This situation is explainable from the orientation of Hollberg (2016) on the preplanning stage and therein frequently used sketching programs. The programs used, Rhino and Sketchup, have an almost exclusive focus on geometry and hence cannot be considered fully functional BIM-authoring tools.

Concluding, the analysis of commercial solutions underlines that BIM-based LCEA for early design stages has shortcomings in interoperability. To be able to remedy these limitations, the critical conversion from BIM to BEM is discussed in depth.

2.6 BIM-based Interoperability for OE Demand calculations

The requirements for a minimalist screening LCEA may be categorized as geometrical and semantic requirements. While most of the semantic data may also be assigned on the LCEA side, with comparatively less effort in time and less susceptibility to human error (Bazjanac 2010), an automated transfer of geometry is indispensable.

2.6.1 Geometric Requirements of LCEA

The core of LCEA or BEM input is a thermal model, including surfaces with orientation, area, divided according to thermal zones. For this division, the theory of space boundaries (SB) (Bazjanac 2010) is the basis for depicting correctly the assumed one-dimensional heat flow through the building elements. Figure 2-16 illustrates how an architectural 3D-geometry is divided according to thermal logic. OE demand software engines set up equations for heat transmission and radiation for surfaces that may establish a homogenous 1-D transmissive heat flow (Bazjanac 2010). Compared to an architectural wall, a change in materials or confining thermal conditions require a division of the boundary surfaces. The resulting thermal surfaces are called "2nd Level Space Boundaries", whereas the architectural surfaces, before the processing, are called "1st Level Space Boundary".

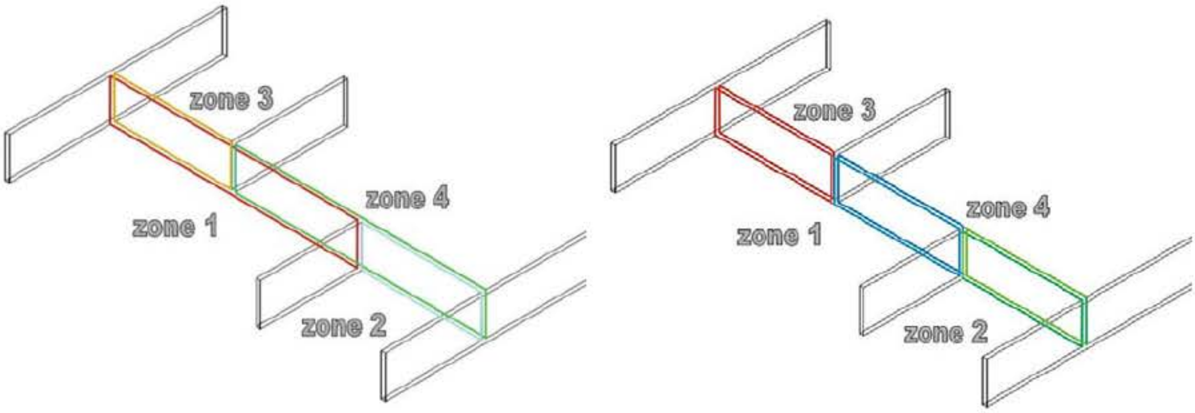


Figure 2-16: 1st and 2nd-Level space boundaries (Bazjanac 2010)

Between the recognized "pairs" of space boundaries (SB), voids may appear where just a 2-D heat flow is possible, e.g. between the bottom and top surface of the walls. In the theory of Bazjanac (2010) those voids are referred to as 3rd, 4th, and 5th level SB. But as the models don't have to be watertight for common, non-dynamical OE calculations, they are often neglected (Rose und Bazjanac 2015) or recompensated by overestimating measuring procedures, like in the DIN 18599.

The theory of space boundaries (SB) foresees a placement of the paired surfaces at the internal of enclosed volumes, as this is consistent with the “paradigm of 1-D heat flow” (Weise et al. 2013). In industrial practice, authoring tools place the 2nd Level SB in the middle axis of building elements, e.g. Revit. The German EnEV requires the external measures for the building envelope, referring to the DIN 18599, see Figure 2-17.

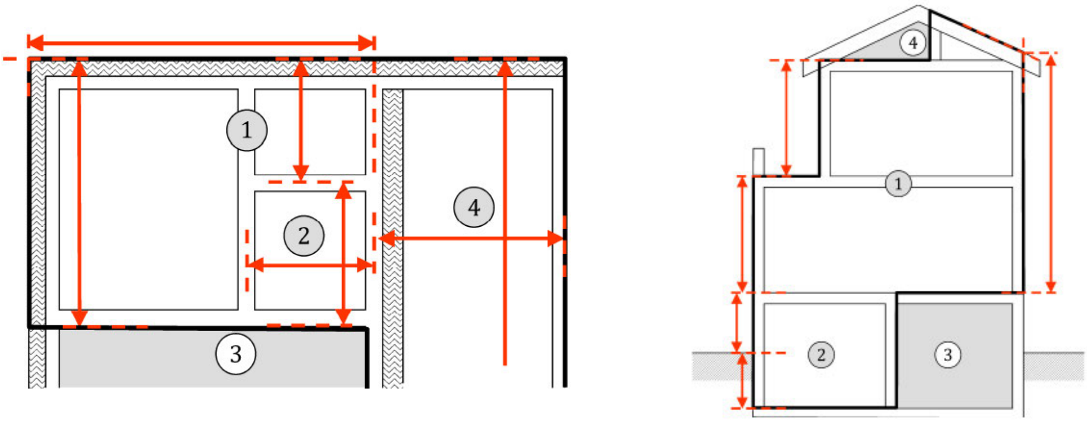
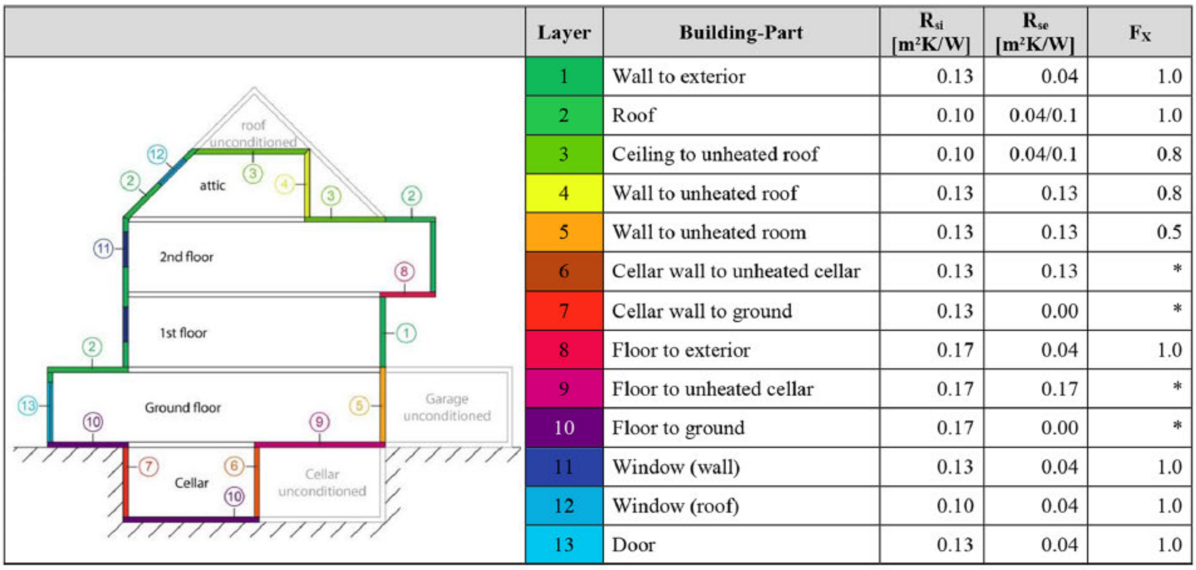


Figure 2-17: Measurement of 2nd Level SB according to DIN 18599

Another ratio required for OE calculations is the gross volume, see section 2.2.1. The gross volume is used as a parameter for ventilation heat losses as well as for an approximation for the net floor area, the functional unit used by the EnEV. The gross volume may be calculated as the product of the gross floor areas according to Figure 2-17, multiplied with the corresponding storey heights. This concludes the geometric information required by LCEA.

2.6.2 Semantic Requirements of LCEA

The 2nd Level SB geometrically store only area and orientation. Semantically, they also need to store a reference to a thermal zone. Further, for OE demand calculation it is also important to distinguish the microclimatic situation of the adjacent building element. For EnEV-conform calculations, 13 different situations of the building envelope are distinguished, see Figure 2-18:



Layer	Building-Part	R_{si} [m²K/W]	R_{se} [m²K/W]	F_x
1	Wall to exterior	0.13	0.04	1.0
2	Roof	0.10	0.04/0.1	1.0
3	Ceiling to unheated roof	0.10	0.04/0.1	0.8
4	Wall to unheated roof	0.13	0.13	0.8
5	Wall to unheated room	0.13	0.13	0.5
6	Cellar wall to unheated cellar	0.13	0.13	*
7	Cellar wall to ground	0.13	0.00	*
8	Floor to exterior	0.17	0.04	1.0
9	Floor to unheated cellar	0.17	0.17	*
10	Floor to ground	0.17	0.00	*
11	Window (wall)	0.13	0.04	1.0
12	Window (roof)	0.10	0.04	1.0
13	Door	0.13	0.04	1.0

Figure 2-18: Categorization of building elements for EnEV-conform OE calculations (Hollberg 2016)

According to the categorization, the U-value of the building element changes as it is influenced by the interior and exterior heat transmission resistances R_{si} and R_{se}. F_x is a correction factor to consider that some parts of a building, e.g. a cellar, are less frequently heated than others and hence also cause less transmissive heat losses, see section 2.2.3.

Knowing the precise requirements posed on interoperability, the method chapter analyzes the documented approaches of data exchange procedures and proposes methodological and technological enhancements.

3 Method

3.1 Problem Formulation and Method

To conduct efficiently a LCEA in the early design stages, the minimal geometrical and semantic data discussed in section 2.6 must be retrieved from BIM-models. Hence, the question is posed how to enable a robust and fast data exchange that accounts for the requirements of this context. Therefore, existing approaches for the BIM to BEM conversion must be researched and analyzed. As approaches in the literature focus exclusively on OE, still an inclusion of EE is necessary.

To answer the challenges addressed in the problem formulation, the method is divided into three main parts: First, three existing approaches for BIM to BEM conversion are introduced and discussed. Second, these approaches are evaluated through a case study that applies process-related and technological criteria. Based on the learnings made in the case study, the third part is the design of an enhanced LCEA data exchange procedure. This procedure shall be demonstrated with a prototypical software implementation and validated for the use case.

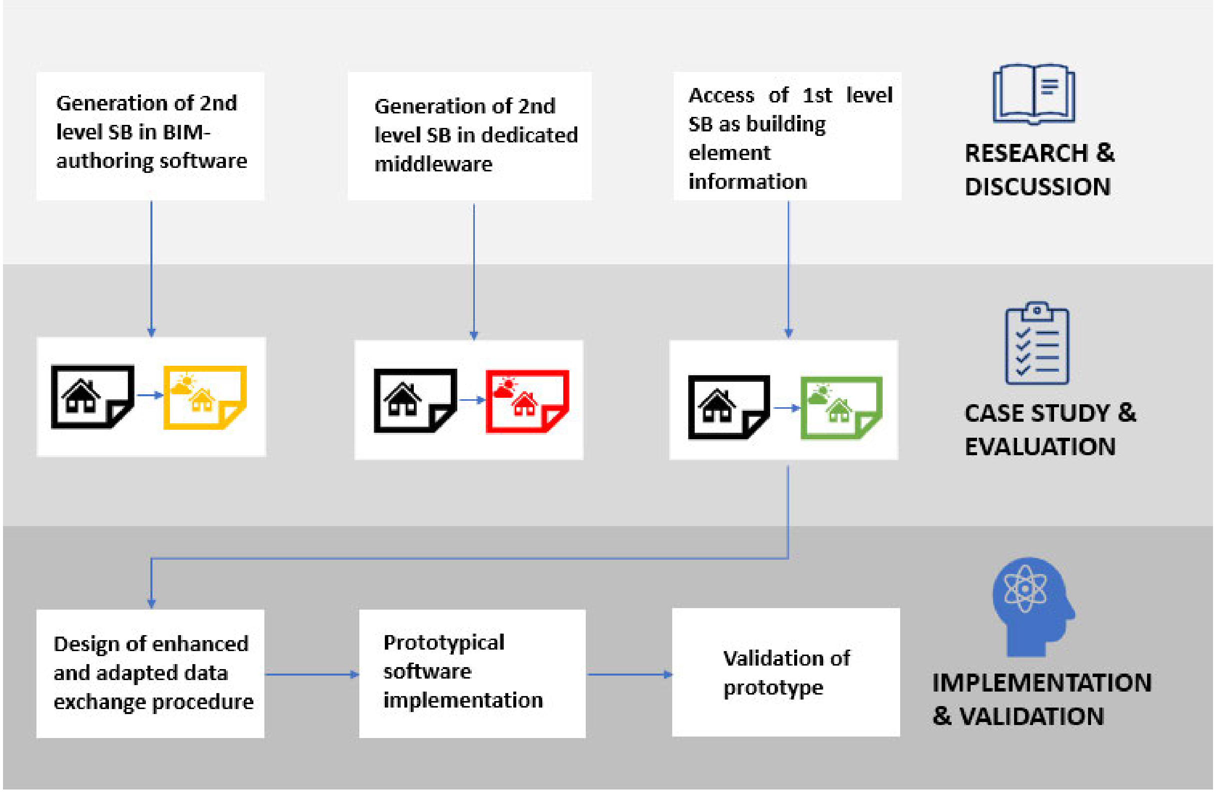


Figure 3-1: Outline of method pursued (own compilation)

3.2 Discussion of existing Approaches of BIM to BEM conversion

Several approaches to an automated data transfer were found to be documented. Thereby, the generation of 2nd Level SB is commonly the crucial source of error in the BIM to BEM process (Rose und Bazjanac 2015). Because of this, the approaches are categorized and discussed as follows:

- Approaches based on 2nd Level SB generated by authoring tools
- Approaches generating 2nd Level SB in dedicated middleware
- Approaches extracting the building envelope geometry without distinction of zones

These three approaches are introduced in the following, in order to systematically evaluate them in section 3.3.

3.2.1 Approaches based on 2nd Level SB generated by authoring tools

To generate the 2nd Level SB from authoring tools may be considered the default way to generate a thermal model from a BIM-model. Therefore, it has been described yet in section 2.5.2. The open exchange formats commonly used are gbXML and IFC, whereby the procedure and data exported is equal.

As mentioned in 2.5.2, the correct export of 2nd Level SB is not a trivial process. Exemplarily, it may be referred to the detailed advise Ecosai Lesosai (2020) give to their customers for the gbXML-export from Revit:

- The modeling of curved walls as polygon traverse
- The modeling of building elements of the building envelope as one-layered systems and assignment of a construction inside the LCEA software
- The replacement of the dormer-family with roof windows.
- Modeling of curtain walls with the regular wall-family and filling it with windows
- The usage of certain window-families
- The modification of certain architectural details (that likely appear at fascias or terraces)

Manipulating an existing BIM-model according to these rules is a very tedious and time-consuming task. Ecosai Lesosai (2020) reported that many users prefer to remodel the entire BIM-model from scratch, what would greatly discourage LCEA in the early design stages.

3.2.2 Approaches generating 2nd Level SB in dedicated middleware

In literature, five algorithms were found that generate the 2nd Level SB based on 1st Level SB exported in the IFC format. They follow the procedure which is illustrated in Figure 3-2.

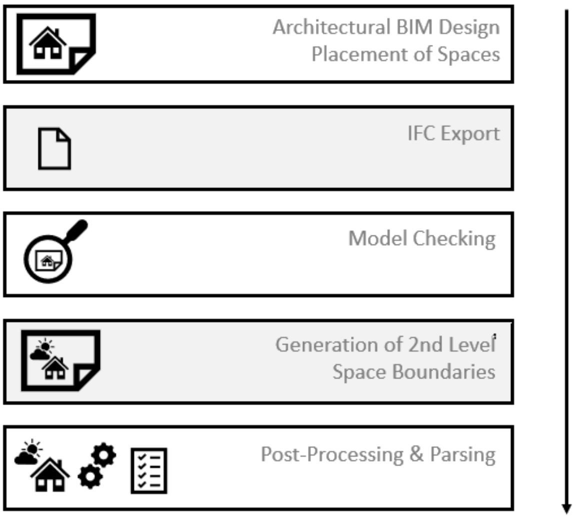


Figure 3-2: Workflows for 2nd Level SB generation in dedicated middleware (own compilation)

An architectural model is created in a BIM authoring tool, wherein then space-objects are placed. Giannakis et al. (2019) give detailed guidelines for the placement of spaces for the software OptEEmAL, confirming that the BIM to BEM conversion is not trivial. The space objects store the 1st Level SB of the confining building elements, which are taken as a basis for the generation of the thermally relevant 2nd Level SB.

Table 3.1: Approaches algorithms generating 2nd Level SB in dedicated middleware

Name	**Year**	**Author**	**Availability**	**Input**
Space Boundary Tool (SBT)	2014	Lawrence Berkeley National Lab (LBNL)	Non-Commercial/ Commercial for payment	1st Level SB in IFC 2x3
Common Boundary Intersection Projection (CBIP)	2015 - 2019	Georgios Lilis, K.I. Katsigarakis	Software built into OptEEmAL (European Commission 2020)	1st Level SB in IFC 2x3 or IFCx4
Ladenhauf	2016	Ladenhauf et. al	Software not publicly available, IP of Austrian Fraunhofer Institute	1st Level SB in not specified MVD
El-Dirabi	2017	El-Dirabi et al.	Software not publicly available	3D solid geometry in not specified MVD (not 1st Level SB!)

Table 3.1 shows the approaches with additional information on their availability and required input data. Exemplarily, the SBT is discussed, which is the only publicly available algorithm, but the oldest one listed. The Space Boundary Tool (Lawrence Berkeley National Laboratory 2013) is a development of the LBNL that was designed to preprocess an IFC 2x3 for the OE calculation software EnergyPlus. The algorithm is described by Rose und Bazjanac (2015) and the software is available open-source with a non-commercial or a commercial license. Today it is not maintained or developed anymore for a lack of further funding. It applies an algorithm based on graph theory where the vertices correspond to 1st Level SB (or a corresponding, external surface) and the edges to their connections – given that between two 1st Level SB a 1D-heat flow is possible. It rotates and projects any possible couple of surfaces to the origin in 2D to analyze the intersecting surface and to crop the identified “pairs”. During the geometric processing, all surfaces are internally represented as a sequence of implicit operations that the explicit geometries of the 2nd Level SB are derived from, in reverse logics. For the export, an IFC model view definition and a Solibri model checker (SMC) constraint set were developed (O'Donnell et al. 2013). The SBT can generate 2nd Level SB for all types of building elements but is not able to process shading on external surfaces or to account for curtain wall elements. Those 2nd level SB can then be exported in the format of the software EnergyPlus or in the input IFC. The other algorithms listed in Table 3.2 are similar in methodology, except for El-Diraby et al. (2017). El-Diraby et al. (2017) did not access 1st Level SB as geometrical raw data, but instead processed the 3D-solid building element geometry, considering adjacent zones. To avoid the use of SB is an idea that the third approach follows as well.

3.2.3 Approaches extracting the Building Envelope Geometry without Distinction of Zones

As discussed in section 2.2.3, the EnEV allows the extraction of the building envelope geometry without the distinction of zones, for most residential building and certain non-residential buildings. In the literature, two publications following this approach were found and are briefly introduced:

Gao et al. (2019) describe a concept for OE demand optimization, with calculations according to the DIN 18599. The focus is the early design stage, whereby real-time feedback shall be given to the user to encourage the frequent usage of the method. Unfortunately, just the concept is published yet and the research project is still ongoing.

Cemesova et al. (2015) describe the extraction and processing of IFC-based geometric data for the low energy design software PHPP (Passive House Planning Institute 2020). For the export of the IFC, Cemesova et al. (2015) had an idea of a fully automated “one click” logic. In the industrial implementation, this idea has been abolished. To process more complex geometries, PHPP imports shared parameters to the authoring tool and demands a manual preprocessing of every building element. Figure 3-3 shows the mentioned implementation of PHPP with the BIM2PH application as freestanding middleware.

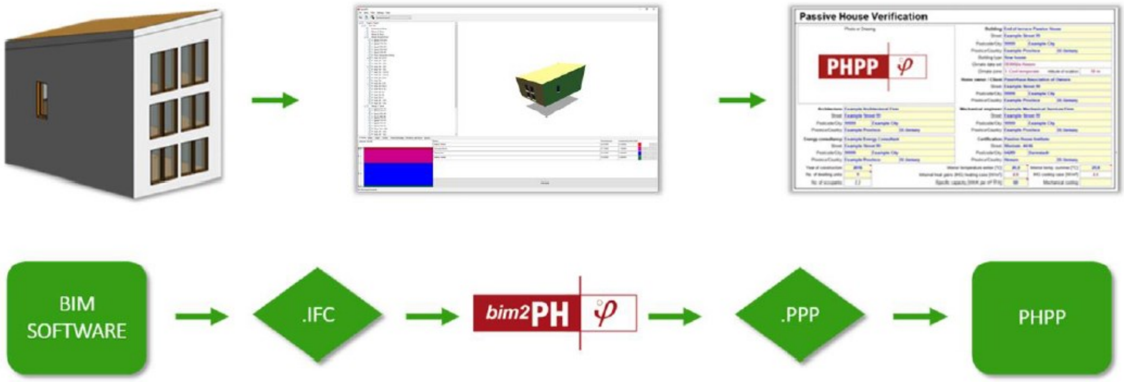


Figure 3-3: Implementation of IFC-based approach of Cemesova (2015) in BIM2PH (PHPP 2020)

The BIM2PH (Passive House Planning Institute 2020) convertor includes a viewer and the possibility to modify all semantic information assigned to the building elements. The PHPP is a complex software intended for the detailed design of passive houses, up to a detailed modeling of thermal bridges or the heat recovery of HVAC. Therefore, the data exchange procedure would have to be adapted to the early design stages and to include EE.

To find out which of the approaches discussed is the most suitable for the use case of LCEA in early design stages, a case study was designed with a commonly expectable BIM-model, designed in two authoring software.

3.3 Evaluation of existing Approaches based on a Case Study

3.3.1 Sample Models

The evaluation was conducted by means of a case study with the following BIM-models, drawn in Revit and ArchiCAD:

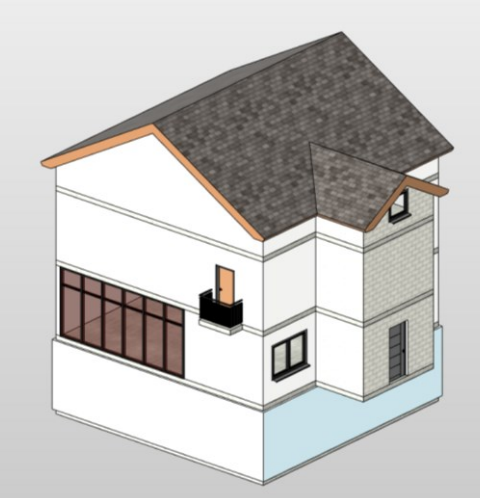


Figure 3-4: Sample building for case study modeled with Revit 2020

Figure 3-5: Sample building for case study modeled with ArchiCAD 23

The building has been modeled in Revit 2020 and ArchiCAD 23 from scratch. To include the most commonly expectable BIM input, the model included the following features covering a variety of geometric and thermal situations.

- An inclined roof with a cross-gable architecture
- Different types of (multi-layered) walls from the component catalogs, including a curtain wall facade
- Different types of windows and doors from the component catalog
- Underground walls and slabs touching the ground
- A balcony
- A porch

The models certainly can be evaluated as superior to models that are averagely expectable in the practice. The foreseen BIM to BEM conversion was taken into consideration from the beginning of the design process, e.g. through attention to error-prone details and an elevation structure that enabled a simple placement of zones.

3.3.1.1 Revit

In Revit, level constraints were applied for all rooms, except for the roof, where an exceeding top offset was chosen. A “general space check” as well as an “architectural BIM model” check were conducted in SMC. In Revit, two types of errors could not be fixed. Firstly, alignment problems were caused by the spaces placed in the roof storey. These were reported not to be aligned with the roof building element. At the sides of the spaces, an insufficient alignment with the curtain wall facade and with the knee walls was reported, see Figure 3-6 and Figure 3-7:

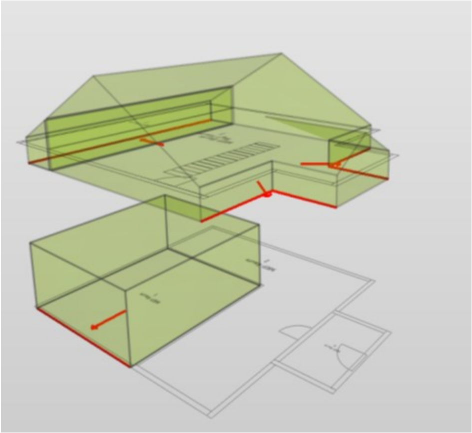


Figure 3-6: Error reporting wrong alignment of a room with curtain wall façade and knee walls

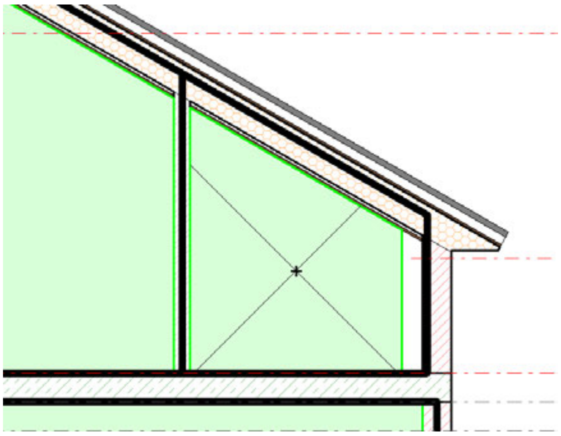


Figure 3-7: Error reporting the wrong alignment in section, not correctable

Secondly, the spaces below the roof were reported to intersect the roof. This error was accepted, as in the model no intersection was visible and correctable.

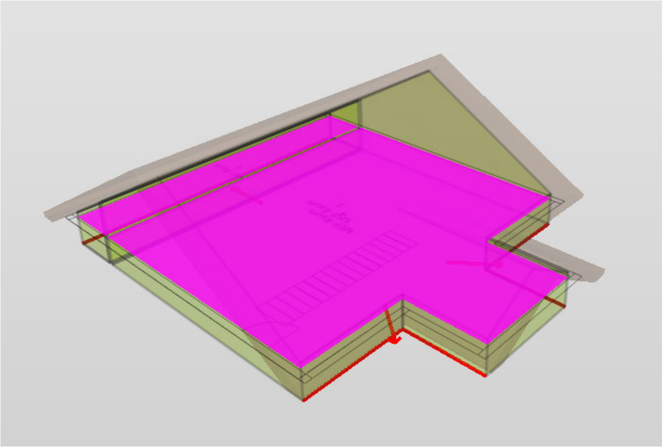


Figure 3-8: Error reporting that the rooms intersect the roof and do not touch a building element on their top surfaces

The error in Figure 3-7, in theory, could have been fixed with a higher knee wall height (>100 cm) or a higher inclination of the roof. As this would not have been a solution applicable in practice, it was not applied.

3.3.1.2 ArchiCAD

In ArchiCAD, the spaces were set using the zone object. According to the documentation of ArchiCAD 23, the zones include the slabs below and above in their offset ranges. For roof elements, the top offset must be set to zero. In comparison to modeling with Revit, two differences may be mentioned in modeling with ArchiCAD: First, the possibility to visualize in 3D the zone-objects. Second, the possibility to modify the objects manually, which is a helpful functionality for complex volumes.

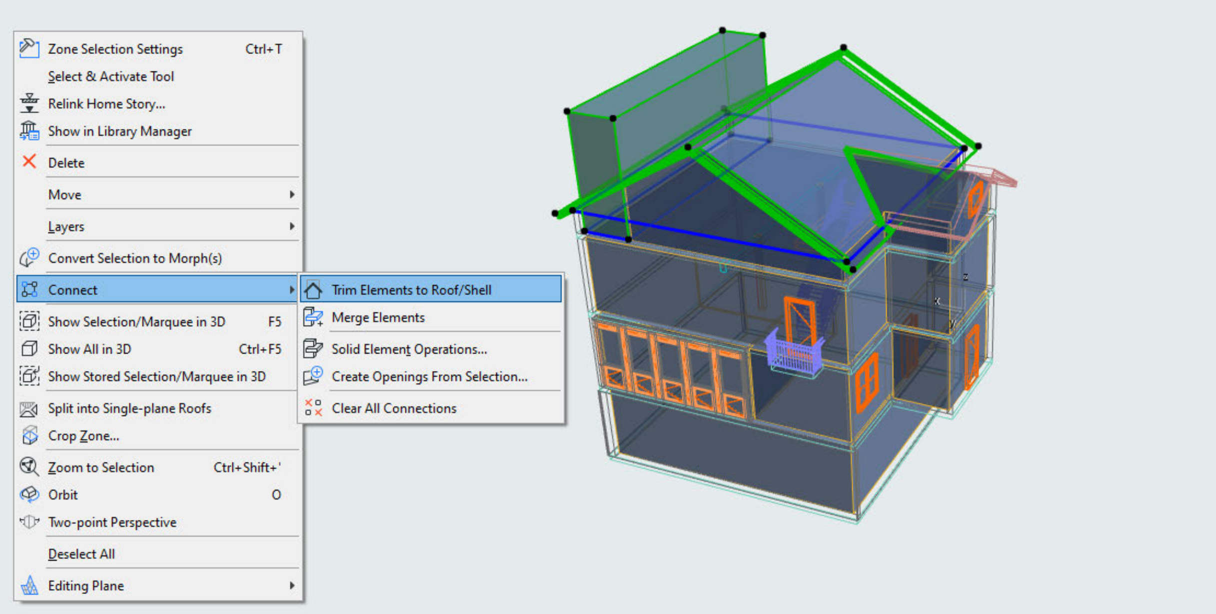


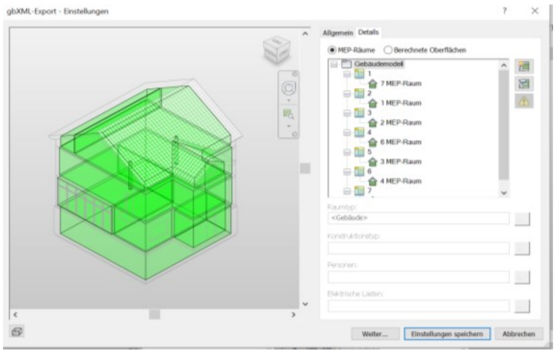
Figure 3-9: 3D-Visualization and manual manipulation of zones in ArchiCAD

For the model exported by ArchiCAD, SMC reported that the walls in the roof storey were not properly linked to the roof placed upon them. As these errors did not cause errors in the conversion to a BEM, they were accepted

3.3.2 Generation of 2nd Level SB inside Authoring Software

3.3.2.1 Revit

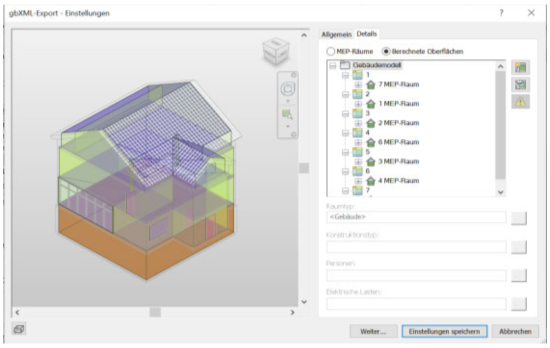
In Revit, the gbXML may be exported with a “one-click” logic or room-based as in Figure 3-10 or Figure 3-11. For the latter, the placed rooms and 2nd Level SB can be shown.



● MEP-Räume ○ Berechnete Oberflächen

Raumtyp: <Gebäude>

Figure 3-10: Visualization of placed MEP rooms in gbXML-exporter



○ MEP-Räume ● Berechnete Oberflächen

Raumtyp: <Gebäude>

Figure 3-11: Visualization of 2nd Level SB calculated based upon placed MEP rooms

The room-based method was found to be preferable due to this real-time feedback. Especially when working with more complex models, it is important support, while the “one-click” method is much more a “black box” when encountering errors.

Figure 3-12: Result of automated generation of 2nd Level SB in Revit using fully automated export functionality

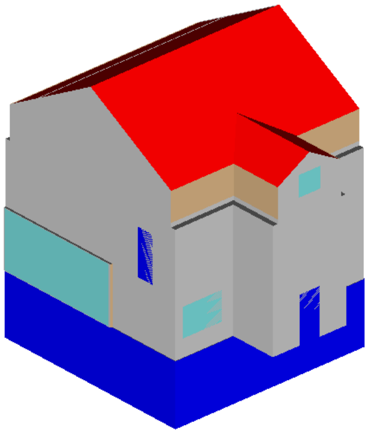


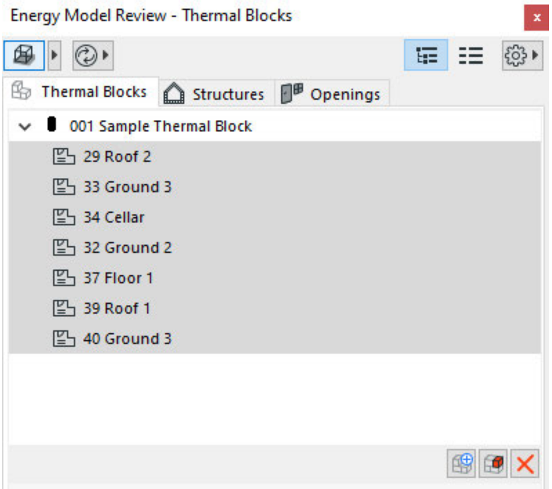
Figure 3-13: Result of the room-based generation of 2nd Level SB in Revit using semi-automated room-based export

Despite the visual support, five iterations were required to achieve the high quality shown in both figures. However, the missing recognition of a part of the roof for the room-based could not be corrected. It is not trivial to set the offset everywhere correctly, to remove clashes, or to clarify poorly modeled details. When processing models that were designed without considering this energetic export, configuring the parametrics of the spaces or to remodel complex details becomes an even more time-consuming and error-prone task.

Summarizing, it may be stated that the generation of 2nd Level SB inside Revit is not a fully reliable process and ideally should be faster and more robust for the early design stages.

3.3.2.2 ArchiCAD

ArchiCAD does not offer a fully automated generation of 2nd Level SB, but only one based upon the placed zones. Therefore, in the so-called “Energy Model Review”, the zones placed have to be assigned to thermal blocks. Instead of starting an energy simulation in the built-in application, one may choose to export the structured data to a gbXML, see Figure 3-14 and Figure 3-15:



001 Sample Thermal Block
- 29 Roof 2
- 33 Ground 3
- 34 Cellar
- 32 Ground 2
- 37 Floor 1
- 39 Roof 1
- 40 Ground 3

Figure 3-14: GbXML-export wizard based on zones in ArchiCAD 23

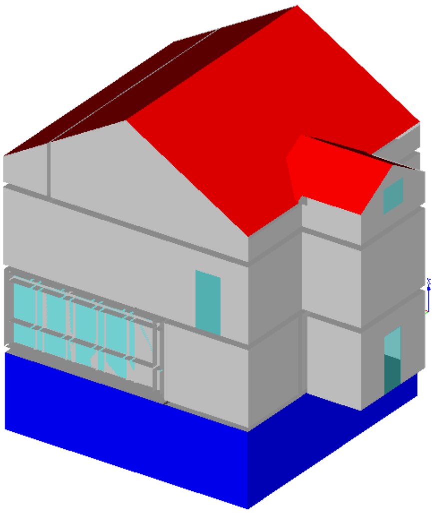


Figure 3-15: Result of the room-based generation of 2nd Level SB inside ArchiCAD 23

Figure 3-15 shows the results of the described process. The results are more accurate than in Revit, due to the more customizable and manually correctable zone placement. Like that, the cross-gable part of the roof and the knee walls could be covered and described as thermal surfaces.

As for the Revit model, several iterations were applied to remove small modeling errors, especially trimming the roof with walls in different modes (contours down or up, different trimming sequences). Errors appearing during the iterations of the gbXML-export were a wrong surface type detected of a part of the roof as in Figure 3-16, entirely missing storeys or one export of just the space-geometry, without any building element surfaces.

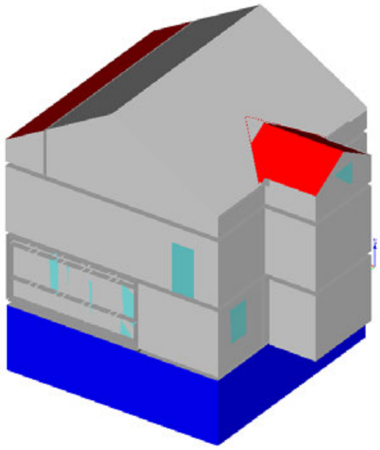


Figure 3-16: Wrong detection of roof-surfaces while trying to obtain correct gbXML-export

It may be concluded that neither in Revit nor in ArchiCAD it is trivial to get a gbXML-export with the entire geometry and the surfaces' correct categorization. Only with several iterations of model correction, export, and visual control, satisfactory results can be obtained, even if the use case was paid attention to during the design process.

3.3.3 Generation of 2nd Level SB in dedicated middleware

The SBT (Rose und Bazjanac 2015) was applied to evaluate this approach because it is the only software available following the approach. SBT may import an IFC 2x3 with 1st level SB and export a processed IFC, including the 2nd level SB.

3.3.3.1 Revit

For the IFC modeled and exported by Revit and its errors reported by SMC, see 3.3.1.1, SBT did not succeed to provide results. The software interrupted at the creation of the output file, without an error message. For an early design stage, in the case study certainly high effort was applied to model the room elements. Smaller errors like the alignment or intersection errors remaining have to be expected in early design, hence this approach has to be considered not applicable for the use case. Even if the curtain wall facade and the roof space were removed, SBT did not produce an output IFC, even though no errors were reported in the SMC constraint set of SBT.

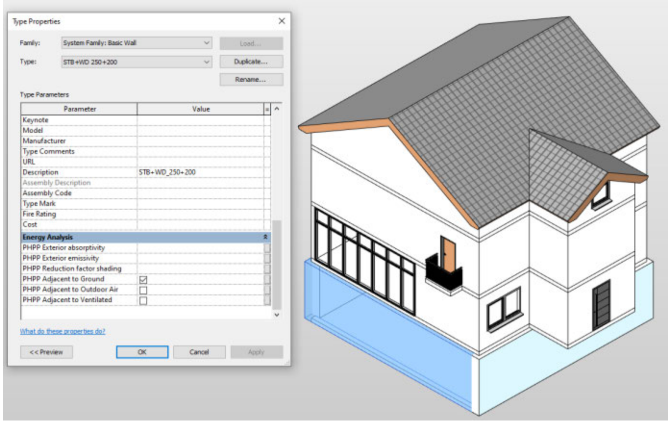
3.3.3.2 ArchiCAD

In ArchiCAD, the export of 1st Level SB is not possible. Hence, the SBT could not be applied on the ArchiCAD-export.

Concerning the approach of dedicated middleware, it may be supposed that the more recent algorithms are more precise and error-tolerant. But as none of them is disclosed and their implementation is out of scope for the thesis, the second approach is concluded with these limited results.

3.3.4 Extraction of Building Envelope Geometry without Distinction of Zones

The third approach tries to avoid the usage of room-elements for geometric operations and accesses building element information. As an example of this approach, the BIM2PH (Passive House Planning Institute 2020) implements the approach of Cemesova et al. (2015). It was applied to the sample model in Revit to evaluate it. As a first step, the parameters required by PHPP need to be imported and assigned with values inside the authoring software. Therefore, the shared parameters are transferred from a delivered Revit template file. The export then could be conducted as recommended in the PHPP-handbook and the file imported in BIM2PH as an IFC 2x3, see Figure 3-17 and Figure 3-18:



Family: System Family: Basic Wall
Type: STB+WD 250+200

Type Parameters

Parameter	Value
Keynote	
Model	
Manufacturer	
Type Comments	
URL	
Description	STB+WD_250+200
Assembly Description	
Assembly Code	
Type Mark	
Fire Rating	
Cost	
Energy Analysis	
PHPP Exterior absorptivity	
PHPP Exterior emissivity	
PHPP Reduction factor shading	
PHPP Adjacent to Ground	☑
PHPP Adjacent to Outdoor Air	☐
PHPP Adjacent to Ventilated	☐

Figure 3-17: Thermal categorization of building elements according to imported parameters of PHPP

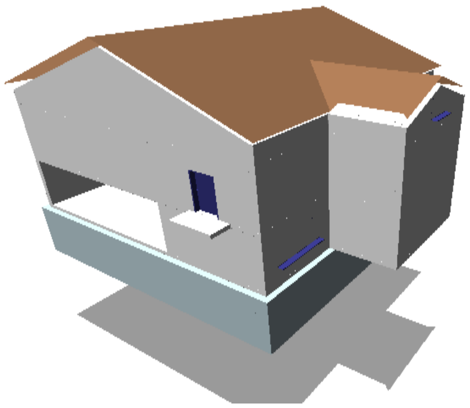


Figure 3-18: Import of exported IFC into middleware BIM2PH

The import via IFC is acceptable, except for the missing curtain wall and two openings that are not displayed by BIM2PH. Inside PHPP, the ".ppp" file generated by BIM2PH may be imported. As during the import of the IFC, error messages are listed. In the surfaces tab of PHPP, every single surface that is listed with its area, construction, and unique identifier. For that, e.g. the area of the curtain wall could be corrected and controlled whether the windows that were not visible in BIM2PH were imported correctly. Of course, this is a deviation from the consistency principle of BIM. However, it enables a more robust data exchange procedure. For minor errors in the early design stages, this can be considered acceptable.

This concludes the practical application of the third approach, wherefore an evaluation can be conducted in the following.

3.3.5 Evaluation of Case Study

3.3.5.1 Evaluation Criteria

All three approaches introduced in section 3.2 in theory could lead to a valid geometry transfer for LCEA. But considering the early design stage and its requirements, the approaches can be weighed up against each other. Therefore, five criteria are introduced: The first three categories refer to technical aspects, the last two take into consideration process-oriented aspects. Every approach is evaluated to meet the criterion negatively (-), neutrally (o) or positively (+). Table 3.2 shows the description of the criteria.

Table 3.2: Evaluation criteria for BIM to BEM conversion approaches

Criterion	**Description**
Robustness	This criterion refers to the sensitivity of the deviations and errors in the model input to the results processed
Achievable Automation	Hereby, the effort to convert a common architectural BIM-model in an early design to a model input for an approach is evaluated
Potential Model Complexity	This evaluates the capacity of an approach to process geometries that do not occur in every building, e.g. curved walls or dormers.
User Training required	A low training effort is important for the early design stage as the user of LCEA and the data exchange is an architect, a non-expert user, who is likely to have little experience in OE demand calculations.
Vendor Neutrality	A high degree of dependency on specific authoring tools is evaluated as negative as it increases the cost of development as well as the prices for users

3.3.5.2 Evaluation of 2nd Level SB generation by authoring tools

As demonstrated in the case study, the generation of 2nd Level SB inside authoring tools may yield good results, if the model is well structured, accurately modeled and rooms are placed perfectly aligned. Therefore, neutral robustness may be assessed. The automation and speed achievable are low, as commonly many iterations have to be trespassed to obtain a valid multi-zone thermal model. In ArchiCAD, a complex building was found to be depictable, in Revit not. But as this approach was the only one found to provide a data basis for multi-zone OE calculations, a high degree of complexity is assessed. The training users need is certainly high, e.g. to correctly place rooms and to model according to the export functionality. The modeler has to think of the use case from the setup of the model and be able to correct reported modeling errors. Finally, the vendor-neutrality is evaluated to be negative, as all of the thermal logic is applied inside an authoring tool. The user depends maximally on the vendor's documentation and support.

3.3.5.3 Evaluation of approaches generating 2nd Level SB in dedicated middleware

As most algorithms require a model free of clashes as well as very accurately placed rooms (El-Diraby et al. 2017), the robustness is comparable to the first method. The degree of automation is low as rooms have to be placed according to multiple rules. Also, no real-time feedback is possible with an intermediary exchange format. Hence, a valid model is only obtained after iterations of exports and checks. It is questionable whether complex geometries may be analyzed better than by means of the authoring software algorithms. Where the rooms were placed correctly, Revit and ArchiCAD provided good results. Regarding user training, this approach is certainly the most demanding. When encountering errors during the application of this approach, scientific publications are less accessible support than forums, documentation, and error messages of authoring tools. Even though a high vendor-neutrality could be assumed, ArchiCAD was not able to export 1st Level SB and contradicted the assumption.

3.3.5.4 Evaluation of approaches extracting the Building Envelope Geometry without Distinction of Zones

The approach of Cemesova et al. (2015) was found to be more robust than the other two approaches. However, a more intuitive checking procedure could increase the robustness, as many parameters have to be assigned in the BIM model. Concerning the speed, the approach is the fastest as the categorization of building elements is intuitive and robust. Due to the very comprehensive and simple modeling guidelines, the approach can account for complex geometries. As discussed in 2.2.1, the QTO without precise zoning is limited to certain conditions. As not all models could be processed with this approach, only a medium complexity achievable is assessed. Further, a medium user training is evaluated. Despite its transparency and comprehensiveness, the user still has to study diligently the data requirements of PHPP from a BIM model to keep an overview of the many parameters imported and transferred via two interfaces. Finally, the third approach is exemplary in vendor-neutrality. It offered templates and detailed guidelines for four different authoring tools and implemented the vendor-neutral converter-platform BIM2PH.

3.3.5.5 Summary

The conducted, textual evaluation may be summarized graphically as depicted in Table 3.3:

Table 3.3: Evaluation of approaches for automated geometry transfer for LCEA

Approach	**Robust-ness**	**Speed/ Automation**	**Complexity**	**User Training**	**Vendor-Neutrality**
Generation of 2nd Level SB inside authoring tools	o	o	+	o	-
Generation of 2nd Level SB in dedicated middleware	-	-	o	-	o
Extraction of the envelope without distinction of zones	+	o	o	o	+

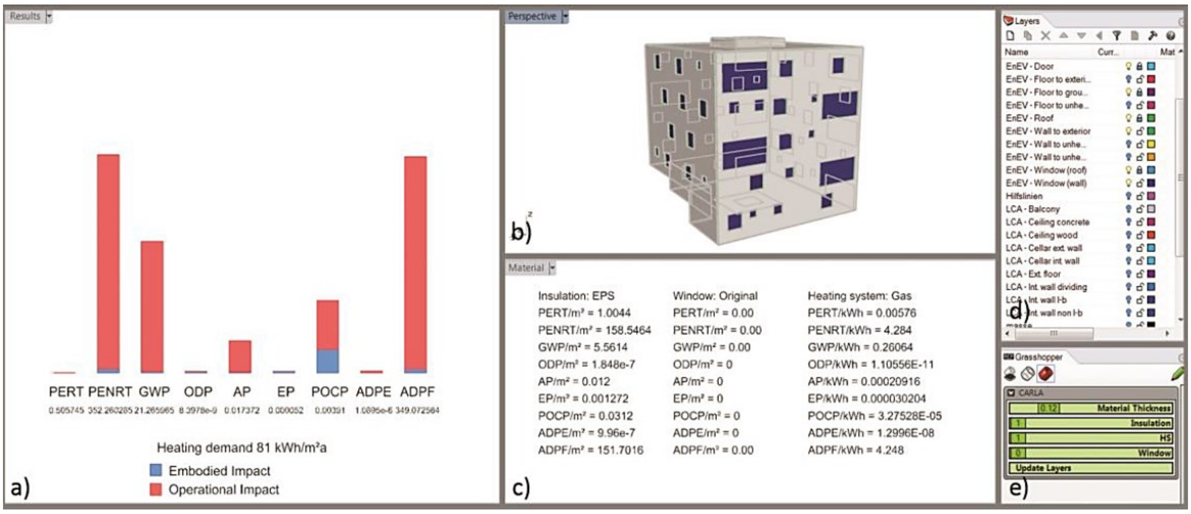
Legend: + positive, o neutral, - negative

For the use case treated in this thesis, the first approach may be the most accurate, but in its effort discouraging to users of LCEA. Approaches of the second type were found to be more scientific contributions than practicable software solutions. The third approach that accesses only building element information is evaluated to fit the best and hence pursued in the design and implementation of an enhanced data exchange.

3.4 Enhanced Data Exchange Procedure

3.4.1 Methodology

To implement any data exchange procedure, a downstream LCEA approach has to be followed. As discussed in section 2.3.2.2, Hollberg (2016) implemented a suitable framework for a design-guiding, iterative LCEA in early design stages. Figure 3-19 shows the user interface he implemented with the sketching software Rhino, including real-time feedback on the results:



Insulation: EPS
PERT/m² = 1.0044
PENRT/m² = 158.5464
GWP/m² = 5.5614
ODP/m² = 1.848e-7
AP/m² = 0.012
EP/m² = 0.001272
POCP/m² = 0.0312
ADPE/m² = 9.96e-7
ADPF/m² = 151.7016

Window: Original
PERT/m² = 0.00
PENRT/m² = 0.00
GWP/m² = 0.00
ODP/m² = 0
AP/m² = 0
EP/m² = 0
POCP/m² = 0
ADPE/m² = 0
ADPF/m² = 0.00

Heating system: Gas
PERT/kWh = 0.00576
PENRT/kWh = 4.284
GWP/kWh = 0.26064
ODP/kWh = 1.10556E-11
AP/kWh = 0.00020916
EP/kWh = 0.000030204
POCP/kWh = 3.27528E-05
ADPE/kWh = 1.2996E-08
ADPF/kWh = 4.248

0.12 Material Thickness
1 Insulation
1 HS
0 Window

Figure 3-19: User interface implemented by Hollberg (2016)

As semantic categorization, Hollberg (2016) differentiated layers to be either part of the building envelope (“EnEV”) or to be exclusively relevant for EE (“LCA”). An important advantage of following the approach of Hollberg (2016) lies in its acessibility in the software CAALA (2020). This is useful as it allows to demonstrate the validity and the application of any data exchange implemented. Therefore, CAALA (2020) was contacted and collaborated within the course of the method.

3.4.2 Technological Choices

Open formats yield two crucial advantages for the user and software vendors. First, it lowers the dependency on specific vendors and increases the competition among them. This ideally results in better products and lower costs for the users. Second, software developers can find better solutions for the interface of one single open standard like IFC instead of dealing with the systems of many different vendors. Cemesova et al. (2015) therefore chose IFC as a data basis for her work. Closed-BIM solutions offer advantages as well, especially considering the use case of LCEA in the early design stages. First, real-time feedback from the LCEA application may be achieved when using a closed-BIM format. Second, the configuration of the data exchange is easier as no middleware for checking or visualization is needed.

Therefore, it was decided to implement a closed-BIM solution. As an exemplary authoring software, Revit was chosen. More specifically, Revit’s object-oriented interface with the programming language C# was accessed. To write interpreted or visual code in pyRevit or Dynamo is certainly well suited for a proof of concept

(Sabatelli 2020), but quickly the procedural logic becomes more difficult to structure, document, and test than a well-designed object-oriented program.

3.4.3 User Workflow Description

Departing from the IFC-based approach of PHPP (Passive House Planning Institute 2020), a manual building element categorizations is pursued. Thereby, a great emphasis is put on communicating to the user the data exchange procedure to avoid the cycles of export, error checking, and correction that are commonly necessary for the room-based data exchanges. Therefore, this workflow with five steps is proposed and partially illustrated in Figure 3-20 and Figure 3-21.

First, the data exchange begins with modeling an architectural model of a BIM model in Revit. Thereby, the modeling guideline has to be followed that every building element has to be unambiguously assignable to one of the thermal layers in Figure 2-18. Exemplarily, the porch of the sample building is unheated, wherefore the floor slabs below and above have to be split. Second, making use of the shared parameter functionality in Revit, the LCEA codes are imported. Potentially, a part of the elements may be automatically categorized at this point. Based on the LCEA code, a visual feedback may be given using view filters and colors in Revit. Thereby, all elements that are relevant only for EE may simply be displayed with the same color to draw the attention of the designer to the correct

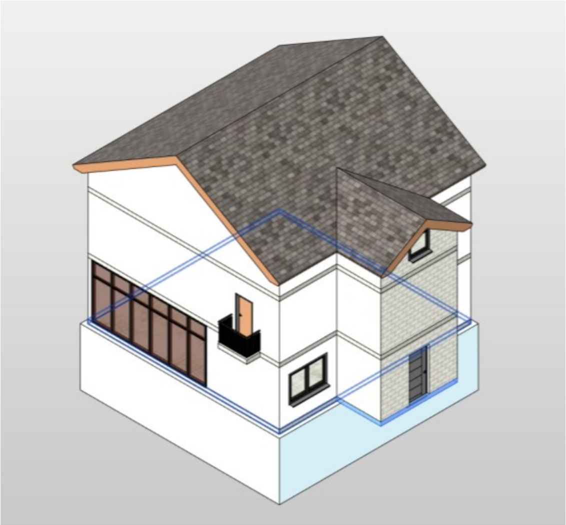


Figure 3-20: Modeling in Revit with elements to be unambiguously assignable to EnEV-Codes(1)

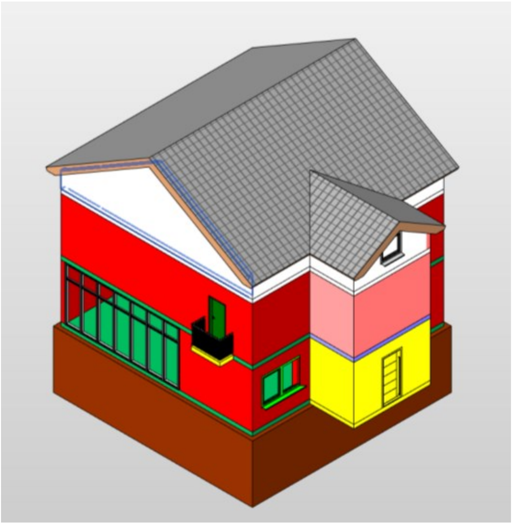


Figure 3-21: Element categorization according to thermal layers in Figure 2-18

Third, to ensure that the model contains the correct quantities to extract, a checking procedure should be implemented to ensure a model input free of clashes, intersections, and with a consistent element categorization. Four, the generated thermal model shall be controllable by the user. Therefore, any commonly used CAD-format shall be provided as an output, e.g. as a gbXML. Finally, the LCEA shall be conducted and intuitive feedback shall be retrieved in real-time.

The implementation of this concept is described in the following sections.

3.5 Implementation

3.5.1 Model Input

The categorization of the building elements is performed by introducing shared parameters. In general, the following layers may be distinguished according to Hollberg (2016):

Table 3.4: LCEA codes imported as shared parameters, grouped to building elements

EnEV-Coding	**Other Elements (EE)**
A01 Wall to the outside air (load-bearing)	B01 Interior floor (heated)
A02 Wall to the outside air (not load-bearing)	B02 Interior wall (load-bearing)
A03 Heated roof	B03 Interior wall (not load-bearing)
A04 Floor against outside air	B04 Wall outside of the envelope
A05 Wall to unheated roof	B05 Column outside of the envelope
A06 Wall to unheated other room	B06 Columns
A07 Wall to an unheated cellar	B07 Roof overhangs
A08 Wall to ground	B08 Interior Windows
A09 Floor against outside air	B09 Interior Doors
A10 Floor against unheated cellar	B10 Balcony
A11 Floor against the ground	B11 Interior floor (unheated)
A12 Window/glass door/curtain Wall	B12 Windows of unheated rooms
A13 Exterior Door	B13 Doors of unheated rooms

Not each of these 16 parameters must be imported distinctly. Some may be deducted logically from fewer others. Some differentiation may be neglected for the early design stage, e.g. the one between A05 and A06. Hence, the parameters listed in Table 3.5 were introduced, grouped to building elements:

Table 3.5: LCEA codes imported as shared parameters, grouped to building elements

Walls	**Slabs**	**Windows**	**Doors**	**Roofs**
isLoadBearing	isHeated	A12 Windows	A12 Glass Door	A03 Heated Roof
CAALA (2020) A01/02: Wall to outside air	A04 Floor against unheated roof		A13 Exterior Door	
A06 Wall to unheated room	A09 Floor against outside air			
A07 Wall to unheated cellar	A10 Floor against unheated cellar			
A08 Wall to ground	A11 Floor against ground			
A12 Curtain Wall				
B – Embodied Energy Only				

As shown in Figure 3-22 and Figure 3-23, view filters are used to give the user feedback on the correctness of his model input.

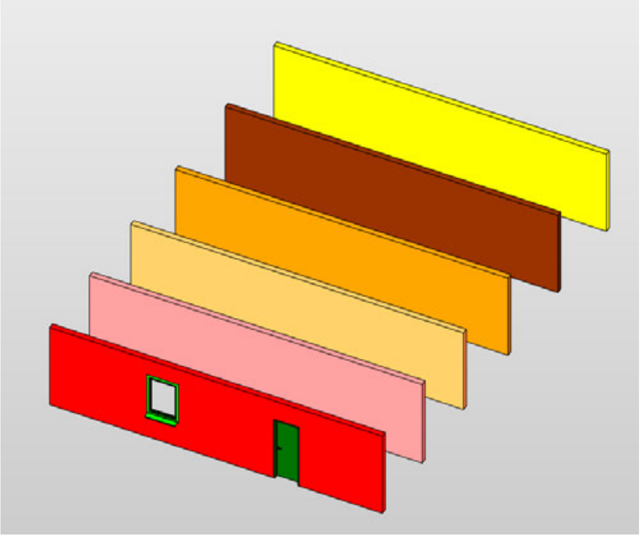


Figure 3-22: Color coding of walls according to Table 3.5 for layers A01, A02, A06, A07, A08, and B-Layers (left to right)

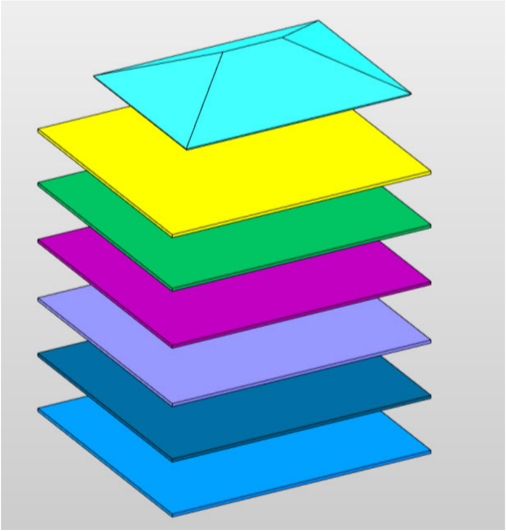


Figure 3-23: Color coding of walls according to Table 3.5 for layers A04, A09, A10, A11 and B01, B11 and A03 (bottom to top)

Beyond this, the view template can indicate inconsistent preprocessing. If a user categorizes a wall as “B” as well as any “A”-Layer at the same time, the wall is displayed grey to communicate the inconsistent input. In the following sections, the software design for the data retrieval procedure is described.

3.5.2 Software Design

The structure of the plug-in is visualized in a unified modeling language diagram (UML) in Figure 3-24. As shown, the code is divided into ten classes. The command-class is the central class, establishing the interaction with the Revit model and coordinating the other classes. Among the other classes, the “Collector”-class is responsible for retrieving the instances of the building elements from the Revit document. The “Converter”-class is responsible for converting the Revit-elements to the internal data structure, which consists out of the semantic and geometrical data indicated in the three non-static classes. To achieve the conversion, the “Converter”-class calls the “LcaCodeParser” and the “GeometryParser” class. The Geometry-Parser does not only extract the area and orientation of a building element but for visualization purposes also its vertices as a polygon loop. Finally, the command class may supply the converted data to the two “Serializer” classes that return JavaScript Object Notation (JSON) and XML-output for CAALA (2020) respectively the Ladybug gbXML-viewer.

Besides the classes displayed in the UML, the classes “XmlAssembly”, “Campus” and “XmlOpening” were created, exclusively for the automated serialization of the gbXML. For clarity, they were left out for the UML. However, the most complex part of the software is the parsing of the explicit geometry, which is discussed in the sections following the UML.

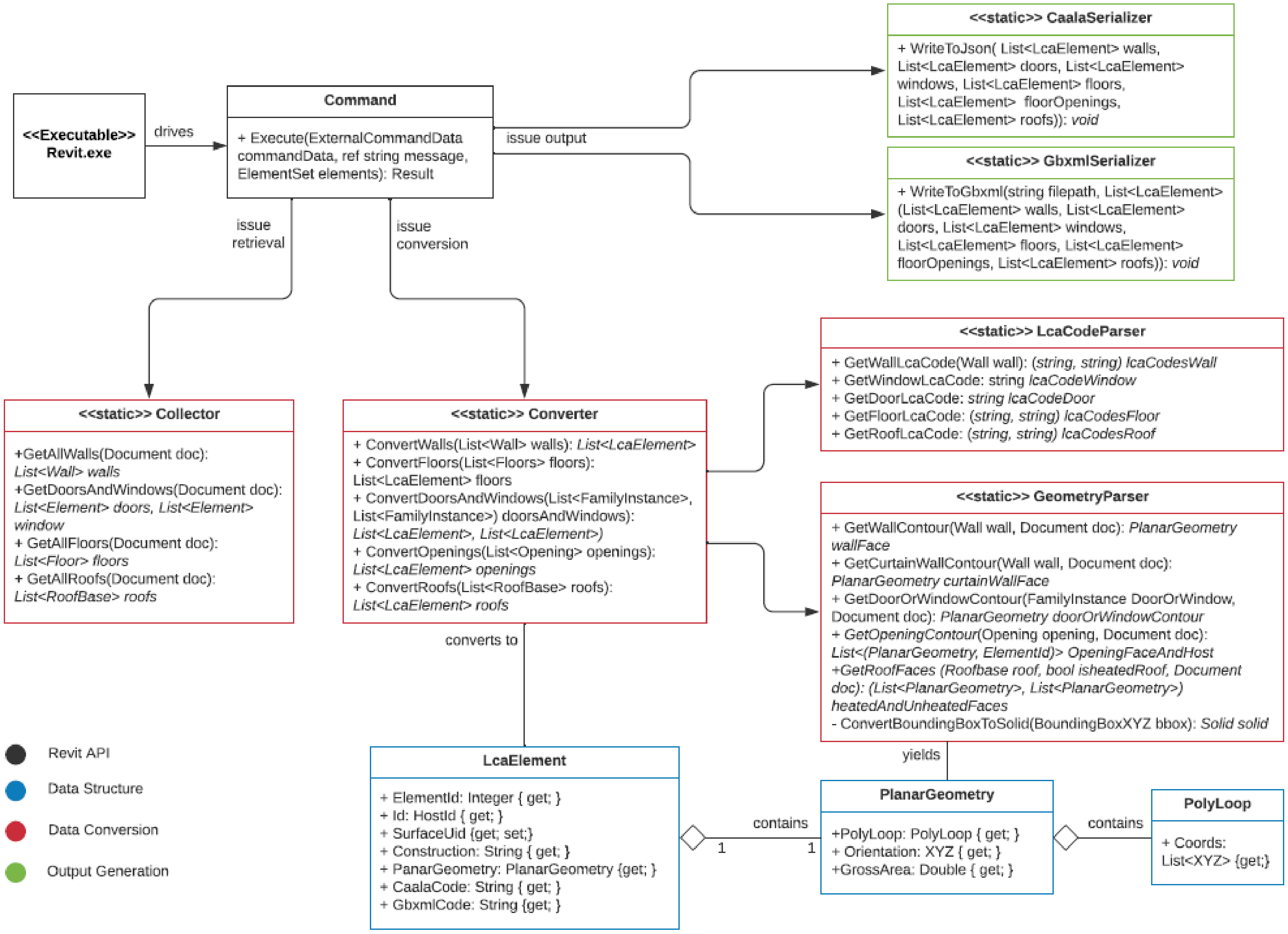


Figure 3-24: UML of data extraction code

3.5.3 Geometry Processing

3.5.3.1 Walls

In the “Collector” class, all instances of walls are retrieved. Thereby, stacked walls are divided into two single “basic” walls. In the “Converter” class, curtain walls are filtered to be treated distinctly in the method “GetCurtainWallContour”. As curtain walls are a combination of multiple 3d-solid elements, it is not possible to extract one face as their external contour. Therefore, a bounding box is created and converted to a solid, whose external face is retrieved. As bounding boxes are placed aligned with the coordinate system, the solid has to be rotated and scaled according to the azimuth-angle of the curtain wall. The applied method is limited to non-inclined and plane curtain walls.

All other walls are inserted in the “GetWallContour”-method. To get the gross area of the wall, this method finds all inserted doors and windows and removes them temporarily with a (later disrupted) transaction. Having temporarily removed the inserts, the solid geometry is retrieved and the external face retrieved as an oriented polygon loop. To calculate the gross area of the wall according to DIN 18599, the head sides of the joined walls have to be added. Therefore, the wall is temporarily translated along and against its location axis. Intersections with other walls are detected, given that they are not coplanar with the wall and not categorized as walls to unheated rooms (“A05”). To calculate the EnEV-conform area, the gross area of the exterior wall face is added to the product of the thickness of the other wall and the minimal height at the intersection. Currently, the method is limited to vertical walls and rectangular joins.

3.5.3.2 Doors and Windows

Doors and windows are retrieved distinctly in the “Collector”-class and processed independently from their hosting walls.

For every “FamilyInstance” that is a door or window, the “GetDoorOrWindowContour” distinguishes whether the insertion element was placed with a cut or with a void. Figure 3-25 and Figure 3-26 show the difference between both placement methods:

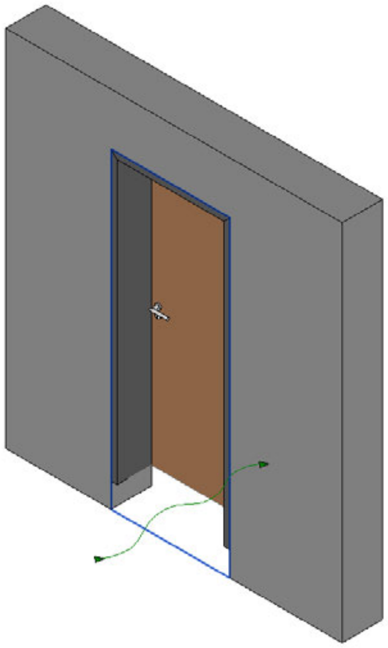


Figure 3-25: Opening defined by a cut

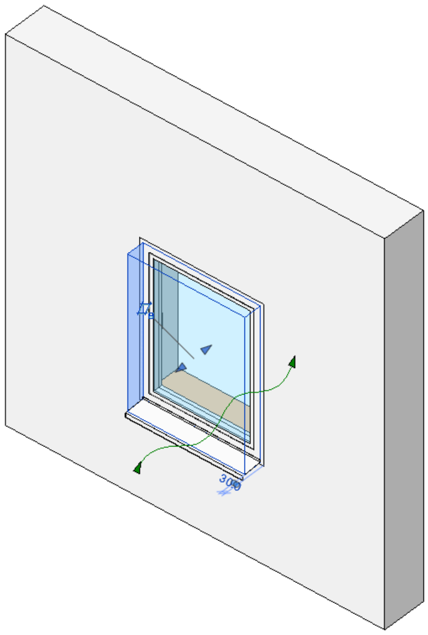


Figure 3-26: Opening defined by a void

For elements placed with a cut, the cut may simply be extracted as a polygon loop. The area and the normal can then be calculated as it is done for walls. If the insertion was enabled by a void, the procedure is more complex. The void blend geometry may only be retrieved from the family-file, which may not be accessible for every model. To establish a robust process, the area of the elements is calculated by the built-in parameters "FURNITURE_WIDTH" and "FAMILY_HEIGHT" of the instance. In case these parameters are invalid or 0, other parameters or default values are assigned. As limitations, this way of calculation is limited to rectangular doors and windows and the accuracy of calculation depends on the conventions followed at the parametrization of the family.

For visualization, an approximation of the geometry is achieved by placing a bounding box around the objects. As for curtain walls, a rotation and a scalation have to be applied. Further, an alignment with the hosting walls is pursued, because gbXML-viewers do not display non-coplanar openings. Currently, the method is limited to vertical walls.

3.5.3.3 Floors and Floor Openings

To calculate the gross area of floors, the upper face is retrieved from the 3D-solid geometry. From the multiple polygon loops defining this face, the external loop defining the floor contour can be determined by sorting all loops. As for windows and doors, openings are distinguished to their modeling technique as cuts respectively as void/shaft openings. The openings modeled by cut may be processed very simply, as they store a reference to their host. The ones defined by a void do not store such a reference, because a shaft may cut several floors. Because of this, the void is transformed into a 3d-solid using its profile and height. Every floor then can be checked for an intersection with the solid. In the case of an intersection, the resulting profile of this intersection is retrieved.

3.5.3.4 Roofs

Without placed room elements, it is not possible to directly recognize the heated part of a heated roof. Therefore, the upward faces were retrieved after the roofs overhangs were temporarily set to zero. For the unheated part of the roof, the orientation is irrelevant. Considering this circumstance, the difference between the area of the entire roof and the heated faces is added as one “B07” element without vertices.

This method is limited to footprint-roofs, while extrusion roofs have not been treated so far. Further, no functionality to consider roof windows has been implemented yet.

3.5.3.5 Gross Volume and Net Floor Area according to DIN 18599

The gross volume of a building is a necessary measure to calculate the ventilative heat losses. Manually, it may be calculated by multiplying the gross measures of every heated floor with their extensions in height, see Figure 2-17. Unfortunately, this process is difficult to automate, as in architectural BIM-models floors may be modeled with console support or covering only a part of the walls when supported by the walls’ top surfaces. The attachment references of walls to a floor are not reliable either. Because of this, it was decided to approximate the gross floor area by multiplying the heated floors with the coefficient 1,04. This coefficient stems from an estimation of a building with a 10 by 20 m extension, a medium wall thickness of 25 cm, and floor slabs that overlap half of this thickness.

These areas are then multiplied with a medium storey height of 2,7 meters to obtain the volume that the floors delimit. The gross volume calculated then may serve as a basis for the ventilative base volume V_e and the net floor area according to the calculation procedures of the EnEV. The users of the software solution shall be communicated that this is just an approximation and recommended a manual calculation if the compliance with the EnEV and the DIN 18599 is desired.

3.5.3.6 Visualization of Geometry Processed via gbXML Format

The “GbxmlSerializer” class serializes the data structure to a minimalist gbXML-format for visualization purposes that the user may use to confirm the correctness of the export. As essential information on spaces, storeys, and materials is not exported, no OE demand calculation software could read it as valid input.

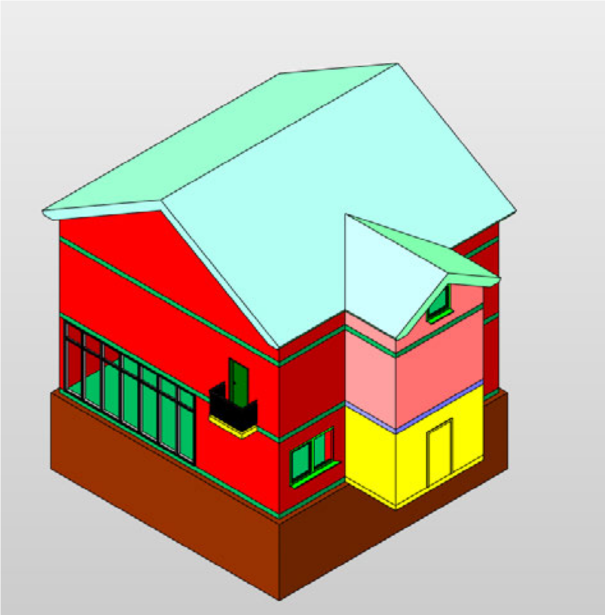


Figure 3-27: Sample Building categorized according to Table 3.5

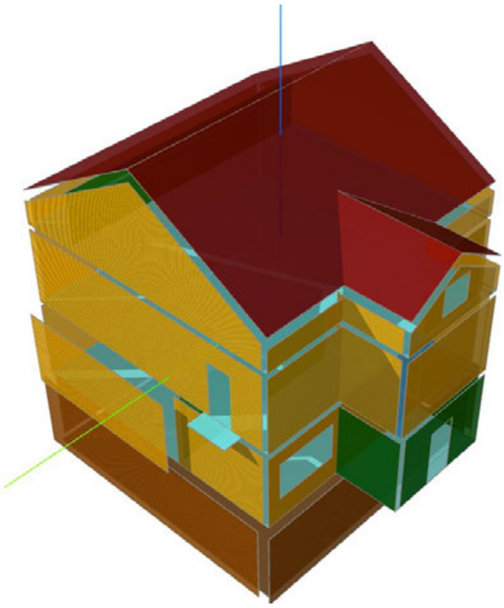


Figure 3-28: GbXML visualization of sample building processed

To understand the visualization, a review of the algorithmics of the “GeometryParser”-class is necessary: First, the voids between the walls appear, because only the geometry of the side faces of the wall is extracted. The area of the head sides of joined walls is considered but does not contribute to the visualization. Second, the curtain wall is not aligned with the other exterior wall. Since the geometry is retrieved by placing a bounding box around the curtain wall, no alignment can be ensured. Third, voids appear between the storeys, as no insulation has been modeled in the sample building.

Of course, the esthetical limitations could be remedied, but without any additional value to the thermal logics. The areas calculated comply with the requirements of the DIN 18599 and EnEV. Given this input, the OE demand may be calculated. For this thesis, the prototype was integrated with CAALA (2020).

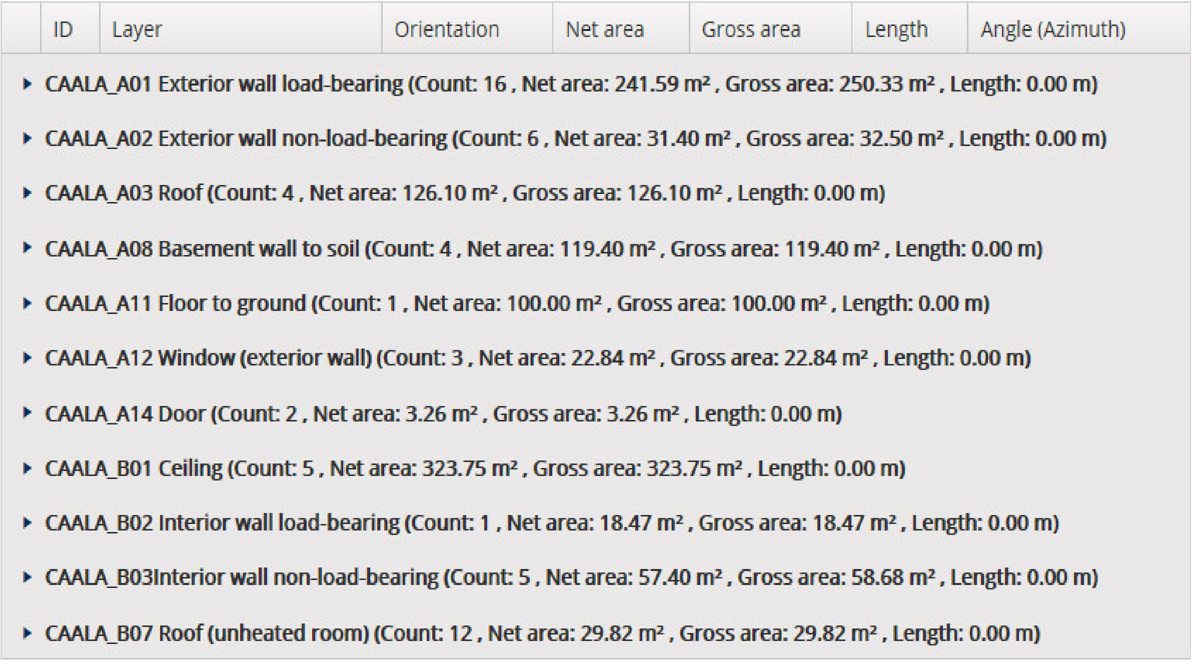
3.5.4 Integration with Downstream Application CAALA

To demonstrate the usability of the solution and the possibility of real-time feedback for LCEA performance optimization, the data was converted to the format of CAALA (2020), which is based on the approach of Hollberg (2016). The format is very similar to gbXML and hence did require little processing of the software's internal data structure. The minimally required properties of the JSON-format are listed in Table 3.6:

Table 3.6*:* Information contained in the JSON-input format for Caala Hollberg (2016)

Surface	**Opening**	**Vertices**
Lca-Code acc. to Table 3.5	Opening Type (Air, Window, Door)	XYZ Coordinates
Unique Identifier of Element	Construction	Unique Identifier of Vertices
Reference to hosted Opening Ids	Unique Identifier of Opening	
Reference to Vertice Ids	Reference to Hosting Element	
Gross Area	Reference to Vertice Ids	
Net Area	Area	
Orientation		
North Vector		

The JSON is sent to a windows forms application that launches an embedded browser and delivers the data to the UI of the software. Inside the UI, building elements are grouped according to the thermal layer. Table 3.6 shows the data schema transmitted to CAALA (2020).



ID	Layer	Orientation	Net area	Gross area	Length	Angle (Azimuth)

▸ CAALA_A01 Exterior wall load-bearing (Count: 16 , Net area: 241.59 m² , Gross area: 250.33 m² , Length: 0.00 m)

▸ CAALA_A02 Exterior wall non-load-bearing (Count: 6 , Net area: 31.40 m² , Gross area: 32.50 m² , Length: 0.00 m)

▸ CAALA_A03 Roof (Count: 4 , Net area: 126.10 m² , Gross area: 126.10 m² , Length: 0.00 m)

▸ CAALA_A08 Basement wall to soil (Count: 4 , Net area: 119.40 m² , Gross area: 119.40 m² , Length: 0.00 m)

▸ CAALA_A11 Floor to ground (Count: 1 , Net area: 100.00 m² , Gross area: 100.00 m² , Length: 0.00 m)

▸ CAALA_A12 Window (exterior wall) (Count: 3 , Net area: 22.84 m² , Gross area: 22.84 m² , Length: 0.00 m)

▸ CAALA_A14 Door (Count: 2 , Net area: 3.26 m² , Gross area: 3.26 m² , Length: 0.00 m)

▸ CAALA_B01 Ceiling (Count: 5 , Net area: 323.75 m² , Gross area: 323.75 m² , Length: 0.00 m)

▸ CAALA_B02 Interior wall load-bearing (Count: 1 , Net area: 18.47 m² , Gross area: 18.47 m² , Length: 0.00 m)

▸ CAALA_B03Interior wall non-load-bearing (Count: 5 , Net area: 57.40 m² , Gross area: 58.68 m² , Length: 0.00 m)

▸ CAALA_B07 Roof (unheated room) (Count: 12 , Net area: 29.82 m² , Gross area: 29.82 m² , Length: 0.00 m)

Figure 3-29: Surfaces of sample building listed in CAALA (2020)

The LCEA optimization now can take place and the learnings transferred to the BIM-model. Therefore, the layers must be assigned a construction from the built-in building component catalog. Thereby, the prototype developed still has two shortcomings: First, no grouping of the building elements according to constructions in Revit has been implemented yet. Walls with different constructions, but the same thermal layer should be optimizable in a distinct way. Second, the assignment of the layers in the future could be automated. However, these functionalities may be easily added as further optimization. The prototypical implementation is validated in the following sections.

3.6 Validation

3.6.1 Testing Methodology

To validate the functionality of a prototype, many different testing approaches exist that differ in focus and scope. For this thesis, it was referred to recommendations of Autodesk Inc (2020). They distinguish four overall testing methods: End-to-end testing, integration testing, compatibility testing, and automated business logic testing. End-to-end testing refers to the validation of software by a third-party “user experience”, in industrial context often a client. Integration testing is the validation of the interoperability of different software coupled in a software solution. Compatibility testing refers to the validation of the functionality of the software in different environments and devices. Business logic testing refers to the validation of the algorithmics of a software solution in different levels of granularities of the different classes and methods, as shown in the pyramid in Figure 3-30. Unit tests thereby refer to single classes and methods, functional tests require the collaboration of different classes and scenario tests motivate users to use the software for use cases it is likely to be applied for.

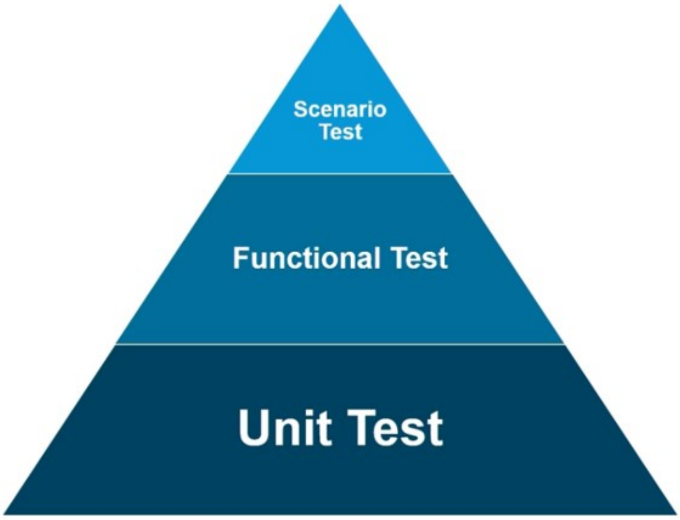


Figure 3-30: Methodology for business logic testing recommended by Autodesk Inc (2020)

For the thesis, integration testing and compatibility testing would not give additional scientific value and hence are omitted in the following. End-to-end-testing is conducted with the professional architect Alexandra Hegmann to retrieve feedback from a potential user of the prototype. Business Logic testing was implemented with the open-source framework xUnitRevit (Cominetti 2020). Thereby, only unit tests and functional tests were applied as scenario testing would have been out of scope for this work.

As the technical validity of the software solution is the condition to validate the workflow in an end-to-end testing, unit testing, and functional testing are described first, in sections 3.6.2 and 3.6.3.

3.6.2 Unit Testing

The unit testing is conducted for the “GeometryParser” class and the “Collector” class, as both methods’ are testable distinctly with unmodified model data. The “LcaCodeParser” class was omitted as it performs a very simple logic.

As an exemplary model input for unit testing, the three models in Figure 3-31, Figure 3-32,o and Figure 3-33 were drawn, containing all types of elements that are processed by the methods.

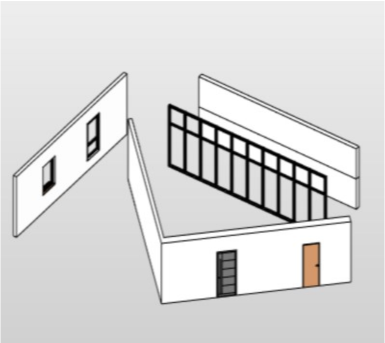


Figure 3-31: "TestWalls.rvt" for unit testing concerning walls(1)

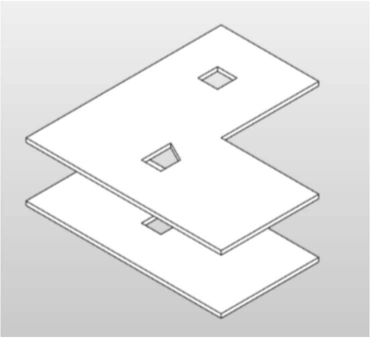


Figure 3-32: "TestFloors.rvt" for unit testing concerning floors (2)

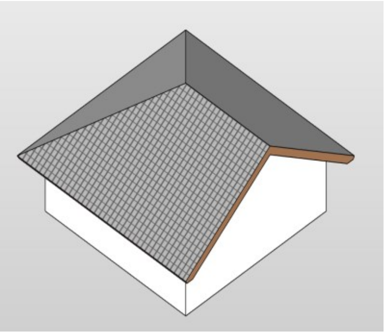


Figure 3-33: "TestRoof.rvt" for unit testing concerning roofs (3)

Model (1) contains five walls from the default catalog of Revit with different constructions, orientations, and types (basic, curtain, stacked). One wall contains two windows, one placed with a void and a cut, another wall a door placed with each a void, and a cut. Two of the walls are joined to confirm the correct calculation of wall areas according to the external measures prescribed in DIN 18599, see Figure 2-17.

Model (2) is built from two floors from the default catalog. One of them is placed as a rectangle, the other as a polygon.

Model (3) is simply a footprint roof supported by four walls forming a rectangle. All sides have an overhang, one side has a gable construction.

Table 3.7: Unit tests implemented to validate methods of the software solution

Method	**Tests**	**Model**
Collector.GetWalls	Number of walls	(1)
Collector.GetDoorsAndWindows	Number of doors and windows	(1)
GeometryParser.GetWallContour	EnEV-conform gross area and orientation of every wall	(1)
GeometryParser.GetWindowOrDoorContour	Area of doors and windows, area, orientation and HostIds	(1)
Collector.GetFloors	Number of floors	(2)
Collector.GetOpenings	Number of openings by type	(2)
GeometryParser.GetFloorContour	Gross area and construction of floors	(2)
GeometryParser.GetOpeningContour	Gross Area and HostId	(2)
Collector.GetRoofs	Number of roofs	(3)
GeometryParser.GetRoofFaces	Number of heated/unheated faces returned, area, orientation	(3)

The expected gross areas, orientations, and host-identifiers were obtained utilizing the Revit-AddIn “Revit Lookup”. Thereby, the area for windows and doors was retrieved from the instance parameters that are accessed by the prototype, see section 3.5.3.2.

All the tests described passed, with an error tolerance of 0,05 m². The only exception was caused by a wall in the test-case asserting the EnEV-conform calculation of wall areas. For the left of the joined walls, a deviation of 0.075 m² was recorded. As this error did not appear for a freestanding copy of this wall neither for any other freestanding wall, it is assumed that this error is caused by the way Revit calculates the intersection with the insulation of the other wall. Considering the context of early design stages, the deviation is certainly acceptable. For the wall with 3 meters of height, it is a proportion of 0,25 % of the area. This is still more precise than conventional manual measurements, e.g. with a ruler, 2D-plans, and excel-tables.

Besides this observation, the precision achieved in the test is not generalizable for windows and doors defined by voids. The exact calculation of their area depends on the right parametrization in the family types instantiated, including the right mapping of the built-in parameters of Revit to the clearance width and height of the openings. Conventions might not abided during the parametrization of Revit family types. Therefore, potential inaccuracies have to be remedied by an appropriate communication of the data requirements to the users.

3.6.3 Functional Testing

As functional testing, the preprocessed sample model from the case study was used, see Figure 3-27. The expected results were calculated accessing the elements’ parameters with the add-in Revit Lookup, while the area calculation method of Revit referenced to the “finish faces”. To explain the orientation of the sample building, the west side is the one hosting the porch, the north side the one hosting the curtain wall. Table 3.8 shows the tests and results obtained:

Table 3.8: Functional tests implemented to validate overall functionality of the software solution

Test	**Expected Result**	**Deviation**
Number of walls, doors and windows	33 walls, 3 Doors, 2 Windows	0,0,0
Number of floors, openings and roofs	7 Floors, 1 Opening, 1 Roof	0,0,0
EnEV-Area of walls and windows/doors being part of the envelope and facing west	10 Walls: 99,7 m² 1 Door: 2,1 m² 2 Windows: 3,9 m²	+0,7 m² -0,05 m² 0,0 m²
EnEV-Area of walls and windows/doors being part of the envelope and facing north	7 Walls: 118,5 m² 1 Door: 2,1 m²	+0,4 m² -0,2 m²
EnEV-Area of all “A02”-walls	6 Walls: 33,8 m²	+1,4 m²
Area of all “B11” floors	2 Slabs: 10,3 m²	0,0 m²
Gross Floor Area and Gross Volume	5 Slabs: 418,4 m² Gross floor area: 435,3 m² Gross volume: 1175,3 m³	0,0 m² -0,2 m² - 0,4 m³

In general, the results are very precise, except for the fourth test calculating the EnEV-areas of all walls categorized as “A02”. This test shows an overestimation of 1,4 m² or 4%, due to a shortcoming of the “GetWallContour” method of the “GeometryParser”-class. The method adds intersections with other exterior walls that are not coplanar to the wall gross area, see section 3.5.3.1. The algorithm omits this addition if a wall is categorized as “A05” (wall to unheated room). This fails due to the way Revit blends walls at nodes, see Figure 3-34. Because the walls are joined in the middle of the thickness of the joined “A02” wall, the algorithm detects an undesired intersection with the wall “A02.” However, an omission of the entire functionality to get EnEV-conform measurements would result in much higher accuracy. Yet, the addition of algorithmics to avoid this error can be subject to further optimization.

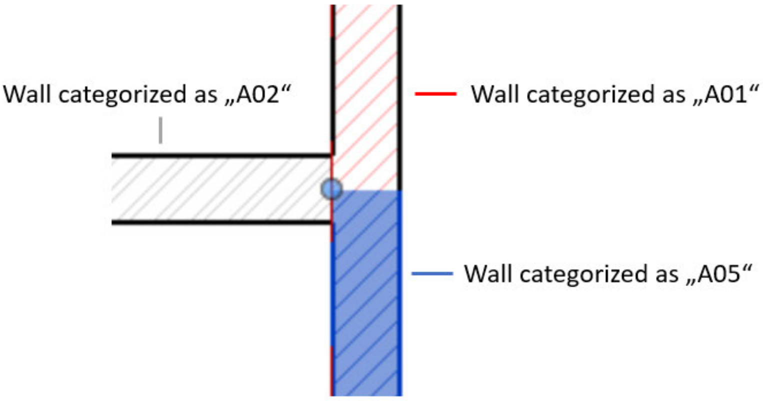


Figure 3-34: Situation causing deviation from EnEV measurement convention in “WallsTest.rvt”

3.6.4 End-to-End Testing

An architect was handed out the prototype to observe whether it is comprehensible and functional in the context it is designed for. According to this context (Hollberg 2018; Schlueter und Thesseling 2009), the architect described herself as very interested in sustainable building and LCA, but without prior professional experience as an energy expert.

The explanation of the workflow and the preprocessing logic took up 20 minutes, including technical requirements. Given this, the architect started to model and categorize a minimalist building with four walls, foundation slab, roof, doors, and windows. After a few minutes, the first LCEA could be conducted without technical problems. However, in a review of the visualization produced and the surfaces listed in CAALA (2020), the orientation of some walls was found to be erroneous. After a correction, the first results could be discussed to optimize the environmental performance of the model inserted. This discussion was influenced by the circumstance that CAALA (2020) assumes envelope properties and HVAC that meet the minimum regulatory requirements in Germany. Departing from this setting, the first learning was that OE then would account for approximately two-third of the model's life cycle PED. The second learning was that this proportion could be lowered drastically by choosing more energy-efficient HVAC, e.g. a heat pump with heat recovery. Only if this is achieved, material optimization started to show a significant impact.

To compare the sensitivity of geometric changes and material optimization, the architect returned to Revit and rotated the model by 90°. Having connected a third time to CAALA (2020), the rotation was found to have a negligible impact. The small window/wall ratio of all sides caused the orientation to be little relevant, although the slightly longer side of the building was now facing the south side. In a final iteration, the change of the extensions to be more equilateral was found to have a higher impact.

The use of the prototype was now reviewed together with the architect, in retrospective. Three interesting aspects were discussed: First, the architect confirmed the simplicity of the workflow that she could learn and apply quickly. For future optimization, she proposed a programmatic model checking and a grouping of building elements according to the material. Second, she reported the transfer of the LCEA results to concrete design decisions to be demanding as a non-expert. However, she found the possibility to do LCEA iteratively with slightly differing variants helpful to become more familiar with the sensitivities. Third, the business context of the early design stage was made the subject of discussion. Despite the interest in the topic, the architect mentioned that the focus of contractors on construction cost makes it difficult to base design decisions on environmental criteria. Most of the competitions she participated in did not incentivize energy efficiency. This naturally lowers the motivation to use the method. A solution to this circumstance could be to better integrate the economic and ecological dimensions of LCEA. Early cost estimations along the entire life cycle (LCC) can motivate contractors to give energy efficient design a higher priority. Similarly, contractors may be communicated that high effort for late design changes may be avoided if they seek feedback on compliance with regulations and certifications.

4 Summary and Outlook

4.1 Summary

The goal of this work was to research and design a robust and fast data exchange procedure between BIM-authoring tools and LCEA software for the early design stages.

The essential demand thereby is the conversion of a BIM model to a building energy model (BEM) that collapses the 3D-solid geometry according to a thermal logic. Three existing approaches from the literature were discussed and analyzed: Two approaches convert BIM to BEM using space elements as geometric input data. They differ in their integration. Algorithms of the first approach are embedded in authoring tools, algorithms of the second approach are available as dedicated middleware. A third approach does not use space objects as basis, but accesses building element information without a distinction of thermal zones. The thermal zones are then approximated in the post-processing, based on empirical data.

A case study showed that the first approach is viable, but time-intensive and requires modeling diligence of the designer. The second approach is limited to scientific considerations, as it is even more sensitive to modeling errors and the implementation is undisclosed for most algorithms. The third approach was found to be the most suitable for the requirements posed, because of its speed and robustness. Based on this approach, a customized workflow was developed. It allows a flexible treatment of commonly encountered BIM-models, an intuitive preprocessing procedure as well as a possibility to retrieve real-time feedback. This was achieved by means of Revit, the C#-Revit API, and the LCEA approach of Hollberg (2016) in CAALA (2020).

As the plug-in stores the explicit geometry, a visualization of the model as a gbMXL is enabled. Besides this visual proof, unit tests and functional tests confirm the algorithmic validity of the prototype. To assess the practical functionality, an architect performed end-to-end testing. She confirmed the benefits of the prototype as didactic support in the optimization of life cycle energy. However, she pointed out optimization potential to leverage the use of LCEA in the industry throughout the entire design process.

4.2 Outlook

4.2.1 Improvement of conventional Procedure

The prototype developed aims to remedy shortcomings of the conventional BIM to BEM algorithms, which are based on room elements. Shortcomings of the currently available algorithms range from their computational inefficiency to their lack of robustness(El-Diraby et al. 2017). Certainly, the missing disclosure of most algorithms might be added as it is a big hindrance for collaborative progress. If these limitations are overcome, the prototype may become less relevant in the future.

Until the technology is mature, applied strategies to avoid remodeling or manual QTOs would certainly be appreciated by practitioners. Given that today automated procedures often fail to process room elements from commonly expectable BIM-models, creative solutions might yield a temporary solution. A viable idea for proficient users could be a semi-automated approach with manual processing of space boundaries in a customized CAD engine.

4.2.2 Optimization of the Prototype

In chapter 3.6, the functionality and validity of the prototype implemented are demonstrated. But being a scientific work, it has a potential for optimization. So far, only planar, horizontally, or vertically aligned geometries were covered. The inclusion of other building element types would contribute to a more granular optimization.

As well there is potential for methodological optimization: A backward communication from the LCEA software to the authoring software might guide the design process more reliably. Schneider-Marin et al. (2020) developed a framework to quantify the uncertainty of EE and prioritize design decisions using a level of development concept. The use of the prototype in the continuity of the design process might be formalized to be compatible with the established industrial BIM-processes (Cavalliere et al. 2019). Abualdenien et al. (2020) proposed a possibility to visualize vagueness inside Revit. The coupling of the vagueness quantification and its visualization would support designers with little experience in energetic considerations.

4.2.3 Application Scenarios

The thesis focused primarily on the data exchange procedure between authoring software and LCEA software. Given a working prototype, the question of its usage naturally arises. There are many application scenarios beyond the design of new buildings. For example, the renovation of the existing stock is a relevant scenario to be accounted for. Besides research focusing on construction physics, influential actors in AEC might give interesting insights. Examples of such actors are asset managers or public contractors with influence on thousands of buildings.

Regarding their economic perspective, the customization of the approach for common regulations and sustainability certifications would certainly increase the readiness of users to apply it. Even more convincing for building owners might be the coupling of LCEA data to life cycle cost indicators (Santos et al. 2019), as economic considerations play a dominant role in nearly every construction project. The integration of many different perspectives is the way to holistic planning that preserves the environment for future generations.

Literature

Abualdenien, Jimmy; Schneider-Marin, Patricia; Zahedi, Ata; Harter, Hannes; Exner, Hannah; Steiner, Daniel et al. (2020): Consistent management and evaluation of building models in the early design stages. In: *ITcon* 25, S. 212–232. DOI: 10.36680/j.itcon.2020.013.

Autodesk Inc (2020): Revit: Autodesk Inc. Online verfügbar unter https://www.autodesk.de/products/revit/overview.

Autodesk Revit (2019): Revit API Docs. Manual to the Revit API. Unter Mitarbeit von Autodesk Revit. Autodesk Revit. USA. Online verfügbar unter https://www.revitapidocs.com/, zuletzt geprüft am 19.03.2020.

Basbagill, J.; Flager, F.; Lepech, M.; Fischer, M. (2013): Application of life-cycle assessment to early stage building design for reduced embodied environmental impacts. In: *Building and Environment* 60, S. 81–92. DOI: 10.1016/j.buildenv.2012.11.009.

Bazjanac, Vladimir (2010): Space Boundary Requirements for modeling of building geometry for energy and other performance simulation. Online verfügbar unter https://www.academia.edu/32704514/Space_Boundary_Requirements_for_Modeling_of_Building_Geometry_for_Energy_and_Other_Performance_Simulation, zuletzt geprüft am 04.06.2020.

Bazjanac, Vladimir; Maile, Tobias; Rose, Cody; O'Donnell, James T. (2011): An Assessment of the use of building energy simulation in early design. Lawrence Berkeley National Laboratory. Stanford, USA. Online verfügbar unter https://www.researchgate.net/publication/264311014_An_Assessment_of_the_use_of_Building_Energy_Performance_Simulation_in_Early_Design?enrichId=rgreq-a5a5c4d6e3bd759fb8e160424165f27f-XXX&enrichSource=Y292ZXJQYWdlOzI2NDMxMTAxNDtBUzo4ODk1NTY3MjMzMTQ2ODhAMTU4OTA5ODI1ODUzNQ%3D%3D&el=1_x_2&_esc=publicationCoverPdf, zuletzt geprüft am 02.07.2020.

Borrmann, André; König, Markus; Koch, Christian; Beetz, Jakob (2018): Building Information Modeling. Technology Foundations and Industry Practice. Cham: Springer International Publishing.

Building Green Inc (2020): LEED. Leadership in Energy and Environmental Design. Unter Mitarbeit von Building Green Inc. Online verfügbar unter https://www.buildinggreen.com/leed, zuletzt geprüft am 23.04.2020.

buildingSMART international (2020): Industry Foundation Classes (IFC) - An Introduction. Online verfügbar unter https://technical.buildingsmart.org/standards/ifc/, zuletzt geprüft am 04.05.2020.

Bundesregierung (2013): Verordnung über energiesparenden Wärmeschutz und energiesparende Anlagentechnik bei Gebäuden (Energieeinsparverordnung - EnEV) Gebäuden (Energieeinsparverordnung - EnEV). EnEv, vom 2013.

Bundesregierung (2019): Broschüre Ökobaudat. Zukunft Bauen. Online verfügbar unter https://www.oekobaudat.de/service/downloads.html, zuletzt geprüft am 25.03.2020.

Bundesregierung (2020): Gebäudeenergiegesetz. GEG. Online verfügbar unter https://www.bmi.bund.de/DE/themen/bauen-wohnen/bauen/energieeffizientes-bauen-sanieren/energieausweise/gebaeudeenergiegesetz-node.html, zuletzt geprüft am 22.10.2020.

CAALA (2020): CAALA. Germany: CAALA GmbH. Online verfügbar unter www.caala.de, zuletzt geprüft am 06.05.2020.

Cavalliere, Carmine; Habert, Guillaume; Dell'Osso, Guido Raffaele; Hollberg, Alexander (2019): Continuous BIM-based assessment of embodied environmental impacts throughout the design process. In: *Journal of Cleaner Production* 211, S. 941–952. DOI: 10.1016/j.jclepro.2018.11.247.

Cemesova, Alexandra; Hopfe, Christina J.; Mcleod, Robert S. (2015): PassivBIM: Enhancing interoperability between BIM and low energy design software. In: *Automation in Construction* 57, S. 17–32. DOI: 10.1016/j.autcon.2015.04.014.

Cominetti, Matteo (2020): xUnitRevit: Speckle Systems. Online verfügbar unter https://teocomi.com/xunitrevit/, zuletzt geprüft am 03.10.2020.

Cooper, Joyce (2014): Life Cycle Assessment (LCA): A Guide to Best Practice. In: *Journal of Industrial Ecology* 20 (4), S. 938–939. DOI: 10.1111/jiec.12424.

DIN EN 15643-2: DIN EN 15643-2:2011-05.

DIN EN ISO 14040, 19.06.2006: DIN EN ISO 14040.

Dixit, Manish K. (2017): Life cycle embodied energy analysis of residential buildings: A review of literature to investigate embodied energy parameters. In: *Renewable and Sustainable Energy Reviews* 79, S. 390–413. DOI: 10.1016/j.rser.2017.05.051.

Ebertshäuser, S.; Graf, K.; Both, P. von; Rexroth, K.; Di Bari, R.; Horn, R. (2019): Sustainable building information modeling in the context of model-based integral planning. In: *IOP Conf. Ser.: Earth Environ. Sci.* 323, S. 12113. DOI: 10.1088/1755-1315/323/1/012113.

Ecosai Lesosai (2020): Lesosai News. Ecosai Lesosai. Switzerland. Online verfügbar unter http://www.lesosai.com/news/news_de_lesosai_2020.html#A8.

El-Diraby, Tamer; Krijnen, Thomas; Papagelis, Manos (2017): BIM-based collaborative design and socio-technical analytics of green buildings. In: *Automation in Construction* 82, S. 59–74. DOI: 10.1016/j.autcon.2017.06.004.

European Commission (2008): A green vitruvius. Principles and practice of sustainable architectural design. Reprint. London: Earthscan.

European Commission (2011): Operational guidance for life cycle assessment studies of the Energy-Efficient Buildings Initiative. EEB Guidance Document Part B: Buildings. Unter Mitarbeit von Bastian Wittstock. European Commission. Online verfügbar unter https://www.eebguide.eu/eebblog/?page_id=659.

European Commission (2020): The OptEEmAL platform. Online verfügbar unter https://www.opteemal-project.eu/news/16/350/the-opteemal-platform-the-ideal-tool-to-design-your-retrofitting-project.html, zuletzt geprüft am 04.07.2020.

Forth, Kasimir (2018): BIM-basierte Ökobilanzierung. Masterthesis. Technical University of Munich, Munich. Chair of computer-based modeling and simulation. Online verfügbar unter https://publications.cms.bgu.tum.de/theses/2018_Forth_Oekobilanzierung.pdf.

Gao, Hao; Zhang, Liwenbo; Koch, Christian; Wu, Yupeng (2019): BIM-based real time building energy simulation and optimization in early design stage. In: *IOP Conf. Ser.: Mater. Sci. Eng.* 556, S. 12064. DOI: 10.1088/1757-899X/556/1/012064.

Giannakis, Georgios I.; Katsigarakis, Kyriakos I.; Lilis, Georgios Nektarios (2019): GUIDELINES for OptEEmAL BIM Input Files. Online verfügbar unter https://www.opteemal-project.eu/files/guidelines_for_opteemal_bim_input_files_v11.pdf, zuletzt geprüft am 07.08.2020.

Green Building Foundation (2020): gbXML. Online verfügbar unter https://www.gbxml.org/, zuletzt geprüft am 18.05.2020.

Harter, Hannes; Singh, Manav Mahan; Schneider-Marin, Patricia; Lang, Werner; Geyer, Philipp (2020): Uncertainty Analysis of Life Cycle Energy Assessment in Early Stages of Design. In: *Energy and Buildings* 208, S. 109635. DOI: 10.1016/j.enbuild.2019.109635.

Hauser, Gerd (2006): Entwicklung eines vereinfachten Nachweisverfahrens für Nichtwohngebäude. Bundesamt für Bau-, Stadt- und Raumforschung BBSR. Online verfügbar unter https://www.bbsr-energieeinsparung.de/EnEVPortal/DE/Archiv/EnEV/EnEV2007/Begleitgutachten/Downloads/DL_Nachweisverfahren_lang.pdf?__blob=publicationFile&v=1, zuletzt geprüft am 07.09.2020.

Hegger, Manfred; Fuchs, Matthias; Stark, Thomas; Zeumer, Martin (2007): Energie Atlas. nachhaltige Architektur. München/Basel, Berlin, Boston: Birkhäuser, zuletzt geprüft am 10.10.2020.

Hollberg, Alexander (2016): A parametric method for building design optimization based on Life Cycle Assessment. Bauhaus-Universität Weimar.

Hollberg, Alexander (2018): Lessons learnt from teaching LCA. Cyprus. Online verfügbar unter https://www.researchgate.net/publication/325381538_Environmental_design_-_Lessons_learned_from_teaching_LCA, zuletzt geprüft am 10.07.2020.

Hollberg, Alexander; Lichtenheld, Thomas; Klüber, Norman; Ruth, Jürgen (2017): Parametric real-time energy analysis in early design stages: a method for residential buildings in Germany. In: *Energ. Ecol. Environ.* 3 (1), S. 13–23. DOI: 10.1007/s40974-017-0056-9.

IBPSA (2020): BIM to BEM: Mission Impossible or Pursuit of hapiness? Online verfügbar unter https://www.ibpsa.us/bim-bem-mission-impossible-or-pursuit-happiness-ibpsa-usa-research-committee, zuletzt geprüft am 01.09.2020.

ISO 10303, 2020: ISO 10303. Online verfügbar unter https://www.steptools.com/stds/step/, zuletzt geprüft am 19.05.2020.

JRC European Commission (2011): ILCD handbook. General guide for life cycle assessment : detailed guidance. Luxembourg: Publications Office of the European Union (Scientific and technical research series, EUR 24708).

Keltsch, Michael (2019): Niedrigstenergiestandard für Nichtwohngebäude. Empirische Potentialstudie zur Realisierbarkeit eines Niedrigstenergiestandards bei Nichtwohngebäuden mit hohem Energiebedarf. Dissertation. Technical University of Munich, München. Chair EnPB. Online verfügbar unter https://mediatum.ub.tum.de/doc/1515033/document.pdf, zuletzt geprüft am 01.09.20.

Kempf, Heike (2011): Verbessertes Modell zur Berechnung des Energiebedarfs zur energetischen Bewertung von Nichtwohngebäuden. Universität Siegen, Siegen. Online verfügbar unter https://d-nb.info/101770645X/34, zuletzt geprüft am 29.08.2020.

König, Holger (2010): A life cycle approach to buildings. Principles, calculations, design tools. First edition. Munich, s.l.: Edition Detail (Edition Detail green books).

Lawrence Berkeley National Laboratory (2013): Space Boundary Tool. SBT-1. Compiler: VS 2010. USA. Online verfügbar unter https://gaia.lbl.gov/interoperability/SBT/, zuletzt geprüft am 04.07.2020.

Lichtmeß, Markus (2010): Vereinfachungen für die energetische Bewertung von Gebäuden. Bergische Universität Wuppertal, Wuppertal. D - Architektur. Online verfügbar unter http://elpub.bib.uni-wuppertal.de/servlets/DerivateServlet/Derivate-1797/dd1004.pdf.

Malhotra, Avichal; Frisch, Jérôme; van Treeck, Christoph Alban (2019): Technical Report: Literature Review concerning IFC, gbXML and CityGML data models for Energy Performance Simulation; 1st.

Nemetschek Graphisoft: ArchiCAD 23. Nemetschek Graphisoft. Online verfügbar unter https://www.graphisoft.de/archicad/, zuletzt geprüft am 14.04.2020.

O'Donnell, James T.; Maile, Tobias; Rose, Cody; Mrazovic, Natasa; Morrissey, Elmer; Regnier, Cynthia et al. (2013): Transforming BIM to BEM: Generation of Building Geometry for the NASA Ames Sustainability Base BIM.

Passive House Planning Institute (2020): PHPP. Version PHPP 9. Darmstadt: Passive House Planning Institute. Online verfügbar unter www.phpp.de, zuletzt geprüft am 03.09.2020.

Paulson, J. (1976): Designing to Reduce Construction Costs. Journal of the Construction Division, 102(4), pp.587–592.

Ramesh, T.; Prakash, Ravi; Shukla, K. K. (2010): Life cycle energy analysis of buildings: An overview. In: *Energy and Buildings* 42 (10), S. 1592–1600. DOI: 10.1016/j.enbuild.2010.05.007.

Rose, Cody M.; Bazjanac, Vladimir (2015): An algorithm to generate space boundaries for building energy simulation. In: *Engineering with Computers* 31 (2), S. 271–280. DOI: 10.1007/s00366-013-0347-5.

Sabatelli, Francesco (2020): VALUTAZIONE DI SOSTENIBILITA' AMBIENTALE CON IL SISTEMA CAALA E APPROCCIO BIM: DATABASE E PROCEDURE VPL. Master Thesis. Università di Bari, Italy, zuletzt geprüft am 04.06.2020.

Sacks, Rafael; Eastman, Charles M.; Lee, Ghang; Teicholz, Paul M. (2018): BIM handbook. A guide to building information modeling for owners, designers, engineers, contractors, and facility managers. Third edition. Hoboken, New Jersey: Wiley. Online verfügbar unter https://onlinelibrary.wiley.com/doi/book/10.1002/9781119287568.

Santos, Rúben; Costa, António Aguiar; Silvestre, José D.; Pyl, Lincy (2019): Integration of LCA and LCC analysis within a BIM-based environment. In: *Automation in Construction* 103, S. 127–149. DOI: 10.1016/j.autcon.2019.02.011.

Schlueter, Arno; Thesseling, Frank (2009): Building information model based energy/exergy performance assessment in early design stages. In: *Automation in Construction* 18 (2), S. 153–163. DOI: 10.1016/j.autcon.2008.07.003.

Schneider-Marin, Patricia; Harter, Hannes; Tkachuk, Konstantin; Lang, Werner (2020): Uncertainty Analysis of Embedded Energy and Greenhouse Gas Emissions Using BIM in Early Design Stages. In: *Sustainability* 12 (7), S. 2633. DOI: 10.3390/su12072633.

Schöndube, Tim (2016): Primärenergiebedarf resultierend aus Herstellungs-, Nutzungs- und Instand-haltungsphase von Gebäuden in Abhängigkeit vom energetischen Gebäudestandard. Technical University of Kaiserslautern. Kaiserslautern. Online verfügbar unter https://www.bow-ingenieure.de/wp-content/uploads/2020/04/2019_Bauphysiktage_Weiterentwicklung-EnEV.pdf, zuletzt geprüft am 02.09.2020.

Soust-Verdaguer, Bernardette; Llatas, Carmen; García-Martínez, Antonio (2017): Critical review of bim-based LCA method to buildings. In: *Energy and Buildings* 136, S. 110–120. DOI: 10.1016/j.enbuild.2016.12.009.

Weise, Matthias; Liebich, Thomas; See, Richard; Bazjanac, Vladimir (2013): IMPLEMENTATION GUIDE: SPACE BOUNDARIES FOR ENERGY ANALYSIS. USA, zuletzt geprüft am 10.07.2020.

Affirmation

Hereby I declare to have written the Master Thesis autonomously. Only the cited sources and means have been used. Verbally or semantically transferred intellectual property I distinguished as such.

Further I assure not to have handed in the Thesis for another examination.

München, 24. Oktober 2020

Vorname Nachname

Lothar Kolbeck