

This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/authorsrights>



Contents lists available at ScienceDirect

## Advances in Engineering Software

journal homepage: [www.elsevier.com/locate/advengsoft](http://www.elsevier.com/locate/advengsoft)

## Formal simulation model to optimize building sustainability

Pau Fonseca i Casas<sup>a,\*</sup>, Antoni Fonseca i Casas<sup>b</sup>, Núria Garrido-Soriano<sup>b</sup>, Josep Casanovas<sup>a</sup><sup>a</sup> Universitat Politècnica de Catalunya, BarcelonaTech, InLab FIB, Spain<sup>b</sup> Universitat Politècnica de Catalunya, BarcelonaTech, SUMMLAB, Spain

## ARTICLE INFO

## Article history:

Received 3 July 2013

Received in revised form 25 November 2013

Accepted 22 December 2013

Available online 25 January 2014

## Keywords:

NZEB

SDL

Simulation

Sustainability

Optimization

Energy demand

## ABSTRACT

In this work, we present a simulation model that makes it possible to find optimal values for various building parameters and the associated impacts that reduce the energy demand or consumption of the building. In the study, we consider several situations with different levels of thermal insulation. To define and to integrate the different models, a formal language (Specification and Description Language, SDL) is used. The main reason for using this formal language is that it makes it possible to define simulation models from graphical diagrams in an unambiguous and standard way. This simplifies the multidisciplinary interaction between team members. Additionally, the fact that SDL is an ISO standard simplifies its implementation because several tools understand this language. This simplification of the model makes it possible to increase the model credibility and simplify the validation and verification processes. In the present project, the simulation tools used were SDLPS (to rule the main simulation process) and Energy+ (as a calculus engine for energy demand). The interactions between all these tools are detailed and specified in the model, allowing a deeper comprehension of the process that define the life of a building from the point of view of its sustainability.

© 2014 Elsevier Ltd. All rights reserved.

## 1. Introduction

In response to the new European directive 2010/31/CE [1] related to the energy efficiency of buildings and the data published by the International Energy Agency [2], which predicts a consumption increase of greater than 40% over the next 2 decades, we propose the use of a simulator that addresses the problem comprehensively from the initial concept design of the building to its eventual demolition for the development of a Net Zero Energy Building (NZEB), see [3] or [4]. During the modeling process, we will consider the energy cost optimization environmental impacts (energy, CO<sub>2</sub>, NOx, etc.) and the economic and social impacts. We can analyze the model year after year to observe how it changes over time. This makes it possible to introduce changes to the environmental model (climate data, the built environment, base model data, etc.).

For the model development, we define 4 basic modules:

1. Climatic and environmental module.
2. Sustainability optimization of the building.
3. Waste produced by the model
4. Energy compensation.

In this project, we analyze the process related to the energy optimization model. The energy needs of a specific building depend on the phase the building is in. We can categorize the building phases as follows:

1. The *initial design* (envelope development). In this phase, the use of parametric architecture [5] to define the shape of the building can be useful. In that sense, the use of a simulation model helps in the definition of some different alternatives that can be considered to analyze, among other parameters, the wind, the temperature, the maximum use of the climate zone, the sun, and the vegetation (using the existing local resources).
2. The *construction process*, which considers the rehabilitation process, the demolition process (in a rehabilitation case), environmental impacts, CO<sub>2</sub> consumption, the economic cost (the reuse of the demolition materials), and so on.
3. The *life of the building*, a process that optimizes the energy demand and consumption of the building by the standards for indoor comfort.
4. The *deconstruction process* and the reuse of the materials.

In this paper, we focus on the simulation model related to the *life of the building*, which facilitates the development of multiple scenarios automatically to determine a feasible solution. For each scenario, we determine a typology based on the shape of the envelope, the climatic zone and altitude, and the best insulation to use

\* Corresponding author. Tel.: +34 93 4017732; fax: +34 93 4015855.

E-mail address: [pau@fib.upc.edu](mailto:pau@fib.upc.edu) (P. Fonseca i Casas).

in response to the need for minimum energy demand, minimum cost and minimum production of CO<sub>2</sub> (taking into account the materials used). This makes it possible to obtain an optimum level of insulation for a specific building to minimize the energy demand.

## 2. State of the art

Conceptually, NZEB “It is a building with a high level of efficiency, where the little amount of required energy should come from renewable sources, so that the energy balance is zero” [6]. Hence, the NZEB are connected to networks and import (or buy) energy to be consumed and export all or part of the energy produced. The expression that defines a NZEB [6] is

$$\sum_i e_i \cdot w_{s,i} - \sum_i s_i \cdot w_{s,i} = E - S \geq 0$$

where  $e$  = exported energy;  $S$  = energy supplied;  $w$  = conversion factors that are applied to each power source ( $i$ );  $E$  = weighted energy exported;  $S$  = weighted energy supplied.

The EU directive 2010/31/CE, with the initiative to incorporate concepts of reduction in the impact on the environment, regulates, in article 9, which on December 31, 2020 the new buildings need to be energy consumption almost null – nNZEB, nearly NZEB – [7].

To know more related to NZEB some interesting works are [3,4,8] or [9].

Today, every building is connected to energy and social networking, which has changed the rules of the game. We now have to address these concepts to complete the cycle (an idea already introduced in Cradle to Cradle [10] and in which we face a new paradigm of design (see Fig. 1)).

We examine the building energy performance using numerous resources for modeling the prototype. The designer spends a significant amount of time designing the virtual model. Only after studying several iterations of the design can a good solution be identified. Without analyzing all possibilities, the designer cannot ensure that the solution adopted is the best, or at least one of the best, solution. Most commercial simulation software packages allow us to calculate the annual energy analysis of the building but do not optimize the process. Neither the results nor the previous

steps, such as the design, construction or final demolition processes, are optimized.

Currently, efforts have been made to develop approaches and methods for assessing buildings [11], methods to integrate daylight and thermal efficiency [12–14], photovoltaic systems [15] and systems of multiobjective optimization for urban design [16] that take various aspects into account when we create an energy optimization system. Some systems address parts of the optimization problem posed above, such as GenOpt [17], an optimization program for minimizing a cost function that is evaluated by an external simulation program, or OptEPlus [18], an application focused on the tertiary sector. In line with these applications, BEopt [19] uses a sequential search technique to automate the process of identifying an optimal design regarding energy use, and Dakota [20], Design Analysis Kit for Optimization and Tera-scale Applications, is a toolkit that is intended to perform an optimization of the design in a multilevel C++ object-oriented framework.

Commonly used languages for programming simulators or optimization tools are Java, Fortran, C++ and Delphi, among others. The use of different languages can cause problems of integration, model definition and understanding among the researchers who come from different disciplines. One of the proposals of this work is the use of a formal language to simplify the model definition and its use.

Using formal languages, such as Discrete Event System Specification (DEVS) [21], Specification and Description Language (SDL) [22] or Petri Nets [23], is without a doubt the best solution to easily integrate and communicate ideas about the model with the other members of the team, improve interoperability with other models (Co-simulation) and encourage collaboration. An example of a recent work that explores the use of formal languages in this area is [24], or [25], that explores the uses of Co-simulation approach to calculate the HVAC (heating, ventilation and air-conditioning) systems in a building.

In this paper, we use SDL as a formalism to define the model behavior and structure. SDL [26] is a visual and easy-to-understand language. It can detail the processes and the procedures that define the model behavior, which allows engineers, architects and researchers to integrate and understand the models easily without learning a specific and more time-consuming programming language.

Defining the model is not enough; it is necessary to perform several calculations. To do this, we identified different calculation engines capable of analyzing the building from an energy standpoint [27], such as DOE-2.1, BLAST, TRNSYS and IES (VE). The use of EnergyPlus [28] with DOE-2.1 and the BLAST calculation engine (a combination that has passed the BESTEST – Building Energy Simulation TEST) [29] gives us the computing power to analyze many factors that influence the energy required by a building. Thus, DOE-2.1 has become one of the most widely used and internationally recognized calculation engines and has been adopted as the base engine in many commercial software interfaces and eQuest [30].

## 3. The model

One of the aims of this research is to be able to describe, as formally as possible, the structure and the behavior of the lifecycle of a building. This goal is crucial in this project because the people involved are diverse and come from many different backgrounds. To address this issue, we use SDL [22] to formally define the model.

The use of SDL language gives us maximum flexibility to integrate new processes and procedures to the system. Using a formal language is an advantage for programming and for enabling non-specialized technicians to access programming to design their

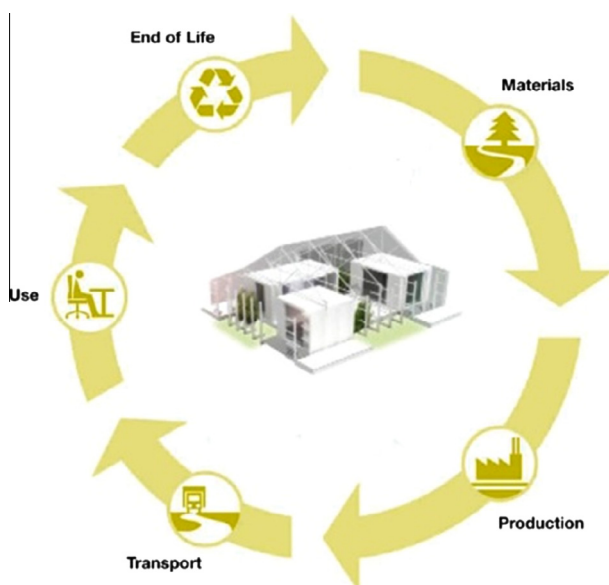


Fig. 1. Cradle to Cradle cycle.

own models and integrate them directly into the simulator much more easily. In addition, the explanation of the model itself is more intuitive, simple and direct than using a programming language such as C. We will teach the programming model to the group so that it will be much easier to understand and detect errors.

### 3.1. Specification and Description Language

SDL Language is an object-oriented formal language defined by the International Telecommunications Union–Telecommunications Standardization Sector (ITU–T) (the Comité Consultatif International Telegraphique et Telephonique [CCITT]) on the Z. 100 recommendation. The language was designed for the specification of event-oriented, real-time and interactive complex systems. These systems might involve different concurrent activities that use signals to communicate. SDL is based on the definition of four levels to describe the structure and the behavior of the models: system, blocks, processes and procedures. In SDL, blocks and processes are named agents. The outermost block, the system block, is an agent itself. Fig. 2 shows this hierarchy of levels.

The different concepts that the SDL language covers are as follows:

- **System structure:** from the blocks to the processes and their related hierarchy.
- **Communication:** signals, communication paths or channels, parameters that can be carried out by the signals, etc.
- **Behavior:** defined by different processes.
- **Data:** based on Abstract Data Types (ADT).
- **Inheritance:** useful for describing the relationships between objects and their properties.

Although a textual SDL representation is possible (SDL/PR), this paper uses the graphical representation of the language (named SDL/GR). More details about the Specification and Description Language can be found in the recommendation Z.100 [22] or on the website [31].

### 3.2. The model structure

The model structure is mainly described by the SYSTEM diagram and the BLOCKS diagrams. In the SYSTEM diagram, we are focused on describing the main elements of our model and the main interactions between the main components (see Fig. 3).

It is remarkable that this diagram defines in a graphical manner the hypotheses that will be considered. In this study, we are following the taxonomy of the hypotheses presented in [32].

The *B1\_Environment* block encapsulates all of the processes related to the environment. As an example, it represents the amount

of radiation that the building receives, the hydrologic conditions, and so on. All of the information that is calculated in this BLOCK is sent to the *B1\_building* block through a file that represents all of the required weather information. This file, “epw,” depends on the climatic zone.

*B1\_Building* is the main BLOCK of the model because it represents the main processes that govern the behavior and the structure of the building. We see this in detail in the following figures.

*B1\_compensation* encapsulates the processes necessary to deliver the energy consumed by the building and to neutralize the environmental impacts (through the process of energy generation for example with solar photovoltaic or CO<sub>2</sub> gas absorption systems).

*B1\_WasteTreatment* represents the processes related to waste disposal. Each of the different blocks of the model is divided into a single process. In Fig. 4, we can see the decomposition of the *B1\_Building* BLOCK into a single process that defines the behavior of the building (*P1\_Building* process).

Following the construction of the SDL structure, we can now define the behavior of the model by detailing its processes.

### 3.3. The model behavior

In this section, we focus on the processes that define the behavior of the building.

The main variables that the model uses are detailed in Table 1.

For the environmental impact variable we are using the categories recommended by the Standards prEN 15643-2 [34], see Table 2, for the social impact we use the categories recommended by the Standards prEN 15643-3 [35], see Table 3.

All of these variables can be taken into account in the model; however, in this paper, we describe only the *Energy Impact* variable. We are focused only in this variable because in this paper we want to explain how the model behaves, and show by an example how we can use this model in a real case. Fig. 6 details this behavior for the *BUILD* state of the *P1\_Building* PROCESS. This is the process responsible for the representation of the entire life of the building, from the design phase to the destruction and recycling phases. Like any other SDL process, it has one or more states. In this case, the building starts in the *DESIGNED* state, representing the fact that a specific design has been selected for the building and that the building process can start. The *BUILD* state represents the fact that the building has been constructed and can be used. The *USED* state represents the fact that the useful life of the building is over and that it needs to be demolished. Finally, the *DESTROYED* state represents the fact that the building has been demolished. Every one of these states has its own calculations and logistics.

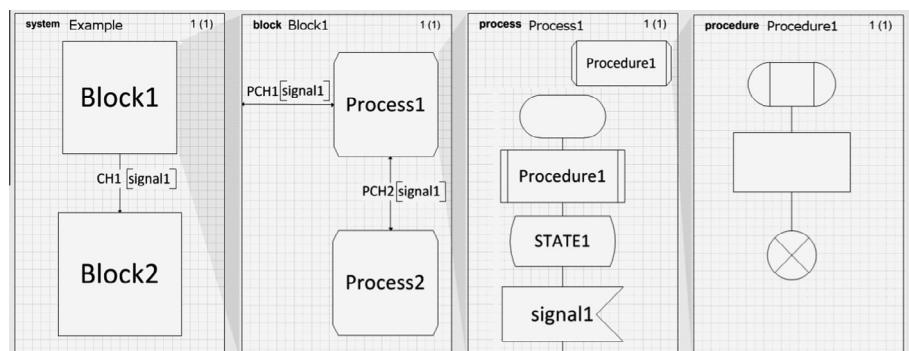


Fig. 2. A structural vision of an SDL model. Four different levels exist.

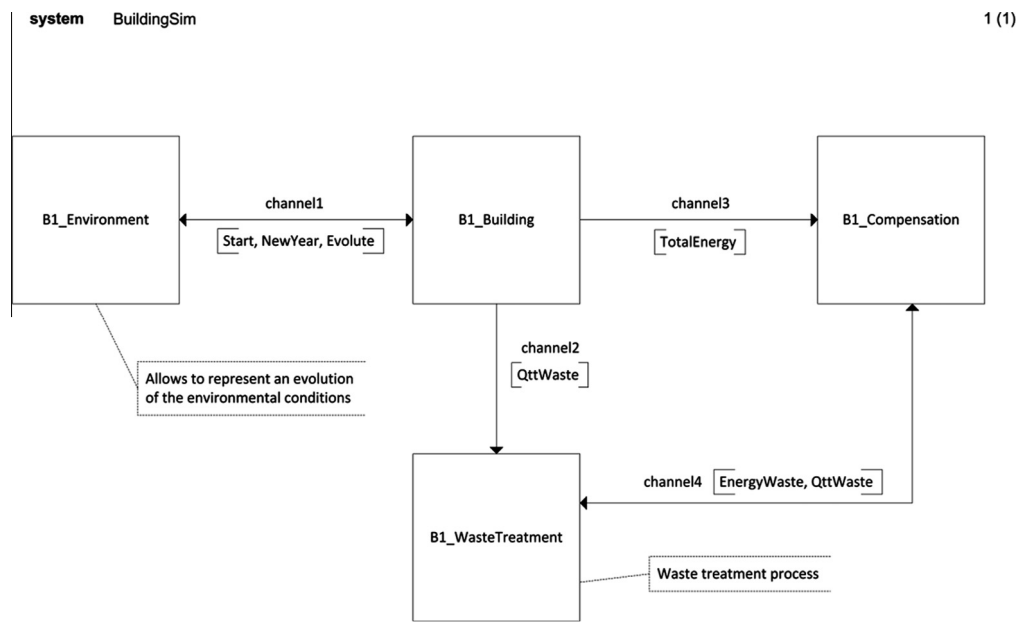


Fig. 3. SDL system diagram detailing the main components of the model to be considered in the study.

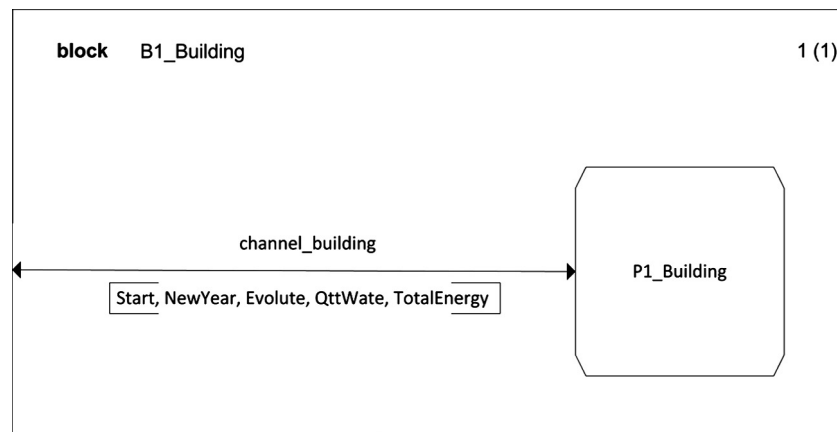


Fig. 4. The B1\_Building block is broken down into a single process named P1\_Building that represents the behavior of the building.

**Table 1**  
Main variables of the model.

Variable	Type	Description
*.epw file (climate file)	Input variable	This file contains the description of the climate that affects the building. In this case, the building is located in Madrid, Spain
*.idf file (model file)	Input variable	This file contains the structure of the building (geometry, materials, etc.)
Energy Impact	Output variable	The energy demand of the model will be determined to minimize the building's energy (energy for heating and cooling)
Environmental Impact	Output variable	Analysis of the environmental impact of materials used (Global Warming, Ozone Depletion, etc.) according to an LCA (Life Cycle Analysis)
Economic Impact	Output variable	Analysis of the economic costs of the LCA process, the materials used in construction and the energy and material demand of the prototype (described in the standard prEN 15643-4 [33])
Social Impact	Output variable	Analysis of the social impacts of the building on its immediate environment (impacts can be positive or negative)

Focusing on the state that defines the construction process (from the *DESIGNED* state to the *BUILD* state) and the behavior of the building while it is used (*BUILD* state), we obtain the diagrams shown in Figs. 5 and 6.

In these diagrams, we use auxiliary variables defined in the DCL sections and PROCEDURES that are declared in the diagram to complete the needed calculus for the model. The entire model is represented in the diagrams, which are complete.

**Table 2**

Environmental impact. Categories used (recommended by the Standards prEN 15643-2). The main database used is the Ecoinvent v2.2 (internationally recognized), always contrasted with BEDEC database (Catalan national database), to prevent slippage.

Environmental impact (BBDD = Ecoinvent)	Unit
Global warming potential	kg CO <sub>2</sub> -equivalent
Destruction of the ozone layer	kg CFC-11-equivalent
Acidification of land and water	kg SO <sub>2</sub> -equivalent
Eutrophication	kg PO <sub>2</sub> -equivalent
Formation of tropospheric ozone	kg C <sub>2</sub> H <sub>2</sub> -equivalent
Radioactive waste	kg
Renewable primary energy use/non-renewable	MJ
Use of freshwater resources	m <sup>3</sup>
Use of renewable/non-renewable (various primary energy)	kg
Resource use recycled/reused (other than primary energy)	kg
Materials for recycling or energy recovery	kg
Hazardous and non-hazardous for final disposal	kg

The second part, which can optionally be skipped, calculates the *building energy consumed* (this part of the model can analyze and

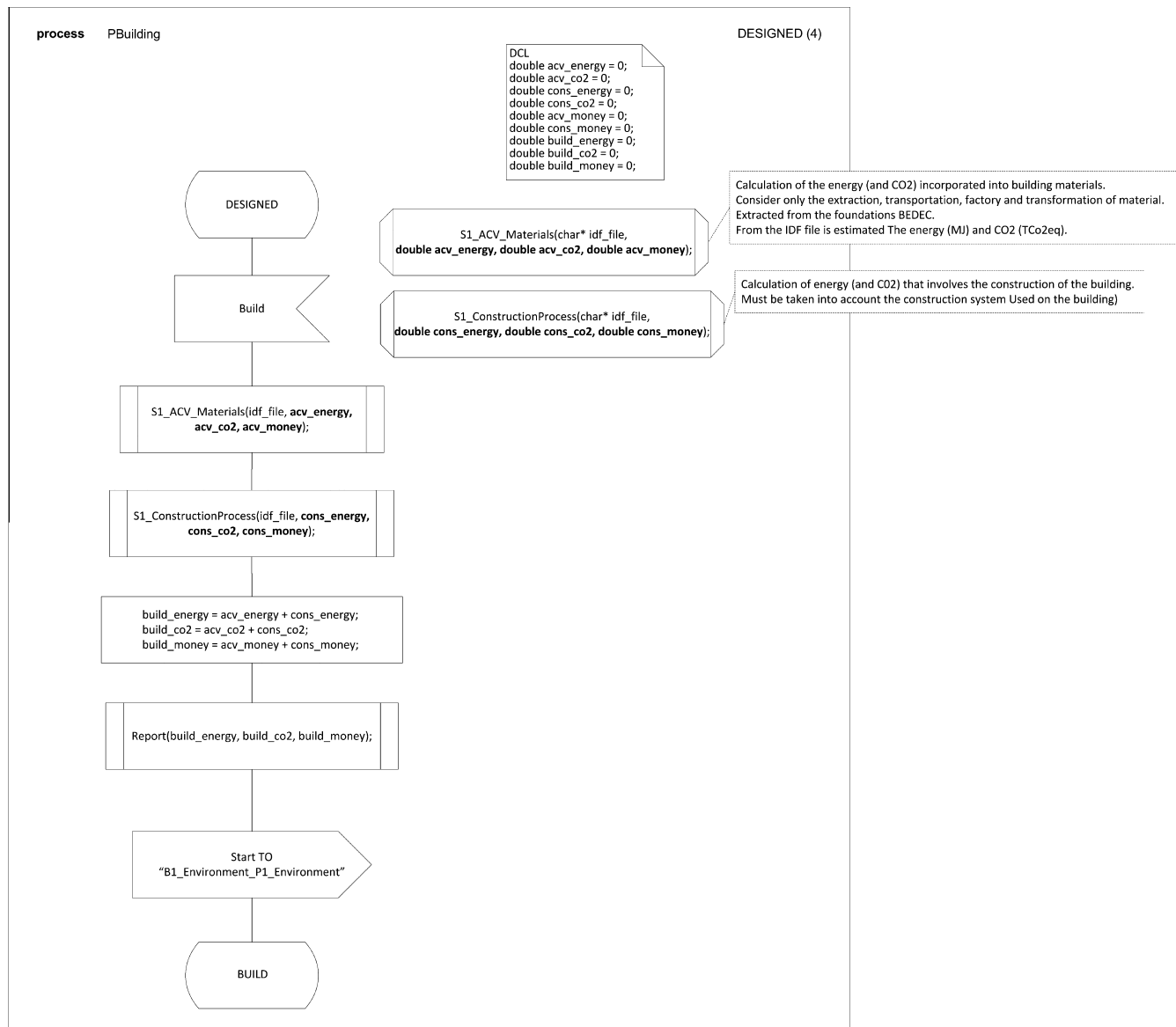
**Table 3**

Social impact. Categories used (recommended by the Standards prEN 15643-3).

Social impact	UNIT
Accessibility	The standard evaluates different impact categories using the methodology contained in prEN 16309
Adaptability	
Health and comfort	
Load the neighborhood	
Maintenance	
Security	

optimize different HVAC systems according to the COP and the performance of each active climate machine so that we know the kW h/m<sup>2</sup> energy consumption required to reach thermal comfort for each situation studied).

We can then more deeply analyze the behavior of the PROCEDURES that use a database to calculate the environmental, social and economic aspects of the materials used and the systems applied to build the model. The *S1\_ACV-Materials* PROCEDURE ana-



**Fig. 5.** This figure details the behavior of the building in the DESIGNED state.



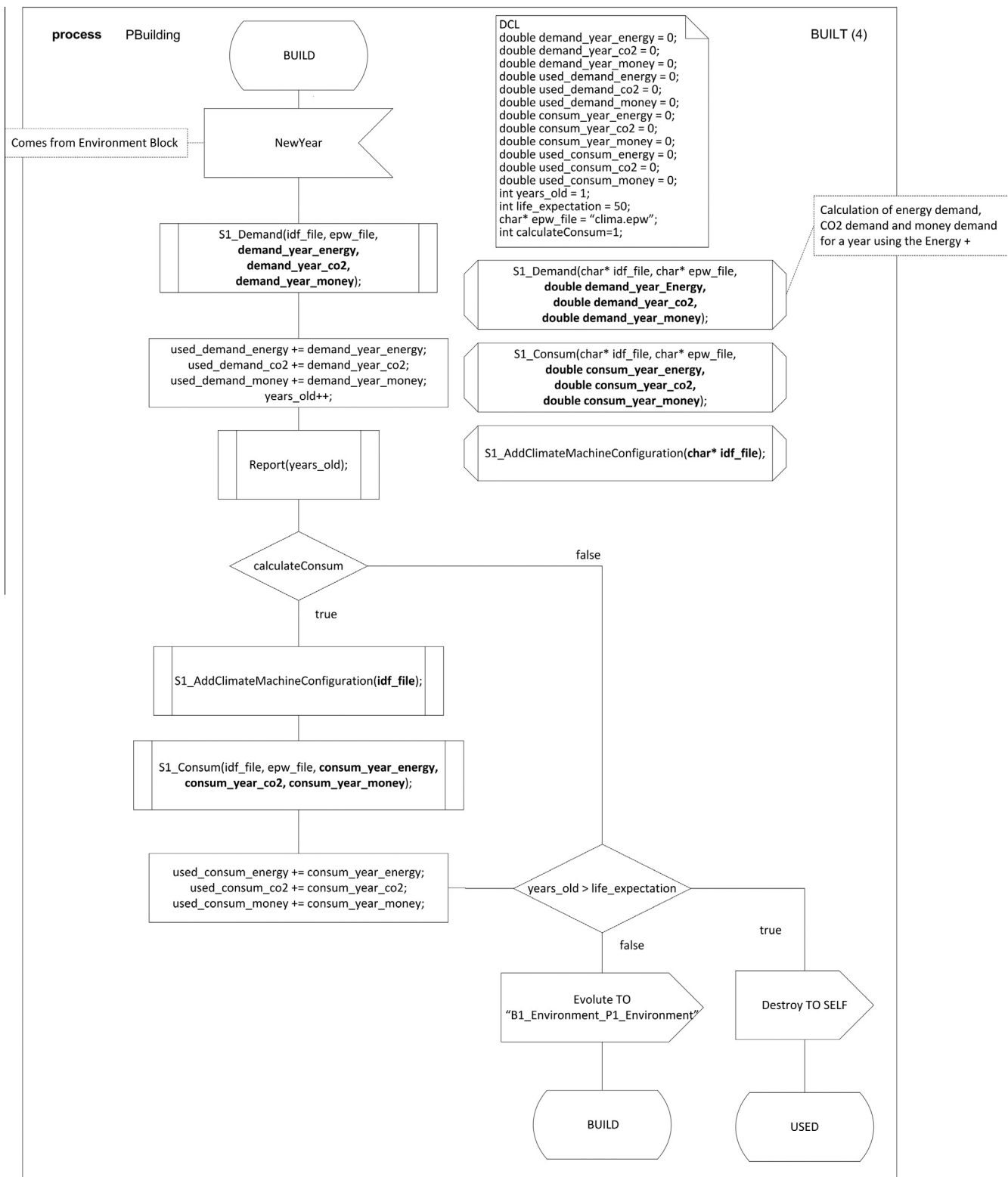


Fig. 6. This figure details the behavior of the building in the BUILD state.

lyzes environmental impacts and the economic and social costs related to materials. The *S1\_ConstructionProcess* PROCEDURE analyzes the construction process (see Fig. 7).

In Fig. 8, we see the *S1\_Demand* PROCEDURE, which takes two input parameters (.idf file and .epw file paths) and produces 4 outputs: the energy and the different demands of the building for one

year (*ImpactAmbiental*, *ImpactEconomic* and *ImpactSocial*). These calculations are made by an external simulator (Energy+), the results of which are incorporated into the model. In this case, the PROCEDURE named *Execute* acts as a link between the SDL model and the external tool Energy+. This procedure is part of the API of the SDLPS simulator.

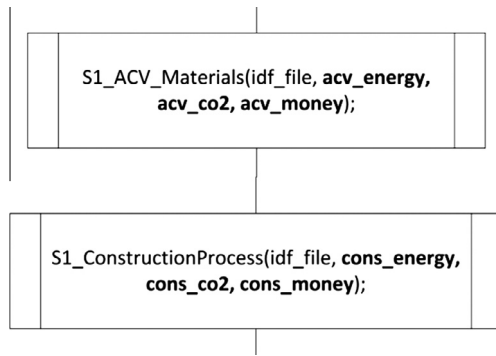


Fig. 7. This figure details S1\_ACV\_Materials and S1\_ConstructionProcess.

#### 4. The case study

The (e)CO project (see Fig. 9) is a prototype of self-sufficient housing designed by the School of Architecture of the Vallès of the Polytechnic University of Catalonia [36] that was been submitted to the prestigious Solar Decathlon Europe competition, SET-2012. The project was being developed by students in collaboration with interdisciplinary research groups, private companies, public administration bodies and other agencies.

It draws on the vast experience gained in the previous contest—Solar Decathlon Europe 2010—in which the project LOW3 (Low Impact, Low Energy, Low cost), submitted by the same research group, (e)CO, won the 1st prize for Architecture [37].

The words “(e) + co” together encompass the current social context. The (e) or “erroba” reinterprets the symbol of our decade. It symbolizes the balance between systems and self-sufficiency. “-Co” involves putting together, incorporating new areas of research, and development and dissemination based on cooperation. As a result, the term “(e)CO” is intended to describe an alternative way to relate to technology, resources and users in the pursuit of low environmental impact solutions in the field of sustainable architecture.

The goal is to develop a self-sufficient housing model based on the principle of zero economic and ecologic footprints. The mechanism to achieve this zero footprint goal is to offer more efficiency, more performance and more comfort using less material, less energy and less money by designing a home in which the organization of the building systems is in keeping with the life cycle.

#### 5. Implementing and executing the model

The implementation of the model is based on software named SDLPS that was developed in the inLab FIB of the [38] Polytechnic

University of Catalonia [39,40]. This tool understands SDL-RT (the code for tasks is defined using ANSI C language) and a set of proposed extensions that will be part of the future release of the SDL standard that simplifies the time management [41–43]. Regarding the infrastructure used, it is remarkable that SDLPS has been built using C++ and C languages. The code related to the model (represented using the C language for tasks and SDL blocks for procedures) is used in the model through a DLL, and the generation of the SDL-XML model is performed through a plug-in in Microsoft Visio®. As a result, the Validation and Verification of the model can be performed mainly by reviewing graphic diagrams in Microsoft Visio®. This dramatically simplifies the interaction between the different parts involved in the project.

As we can see in Fig. 10, the simulator shows the SDL diagrams of the model that are going to be simulated. The system uses an XML representation of the SDL model obtained from Microsoft Visio® working with the SanDriLa© plug-in.

Because we are using this infrastructure, no specific implementation for this project is performed. This simplifies the verification process needed in every simulation project, see [44]. In Fig. 11 is presented the modeling process [45], the simplified steps required in a simulation project are represented in red as a result of using this methodology.

The validation of the model is performed through the SDL representation of the model. As a result of the modular and graphical structure of the language, all of the actors involved in the project can participate in this validation, despite their expertise using any given programming language. The verification is assured because the tool understands SDL diagrams. This methodology assures the complete and unambiguous formal definition of all the elements involved in the simulation model. Thanks to the modular structure of the language the model can be completed with other simulation models that can be used to improve the solutions obtained by the simulator. As an example the model can include the representation of the distribution and the treatments of the garbage generated during the building process, including a model that specifically details this. This holistic representation of the model leads to a complete solution for the problem we want to solve. Other interesting feature of using a formal language is that the model is not tool dependent. In the case of SDL this is especially interesting because is an ISO language. This means that several tools can be used to implement the final model. This makes that the selection of the tool depends on the project requirements and not on the model definition. Finally, due to the completeness and unambiguity of SDL an automatic implementation of the simulator is allowed. Several commercial tools, like PragmaDev [46] or Ciderella [47], support this interesting feature that simplifies the verification (assure that the implementation of the model is

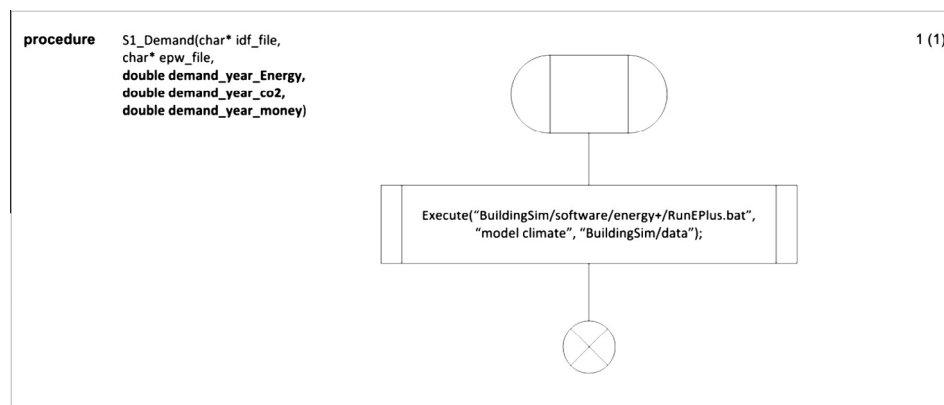


Fig. 8. Procedure S1\_Demand. This procedure makes it possible to call an external simulator that calculates the demand. Currently, we are using Energy+.



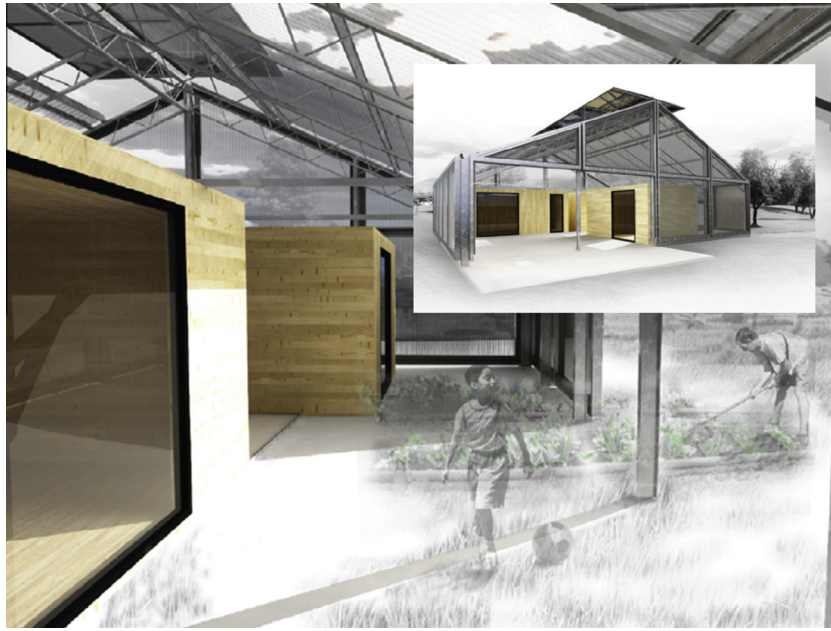


Fig. 9. ECO 2012 solar decathlon building.

correct). Also some techniques exist in order to transform SDL to C++ like [48]. In our case we use SDLPS because it allows a distributed execution of the model and the reuse of legacy simulation models and the integration of other simulation engines in a single tool [49].

To define several scenarios that we will compare, SDLPS makes it possible to define the experiments. For each one of these experiments, a native SDLPS plug-in generates an idf (with the data needed according to the experiment number) and puts this file in a

directory that contains all of the other data needed to execute the experimental scenario.

The conceptual structure of the architecture is represented in Fig. 12.

## 6. Validation of the model

To ensure that the model is validated, we must ensure that the hypotheses used are correct. Several techniques exist to validate a

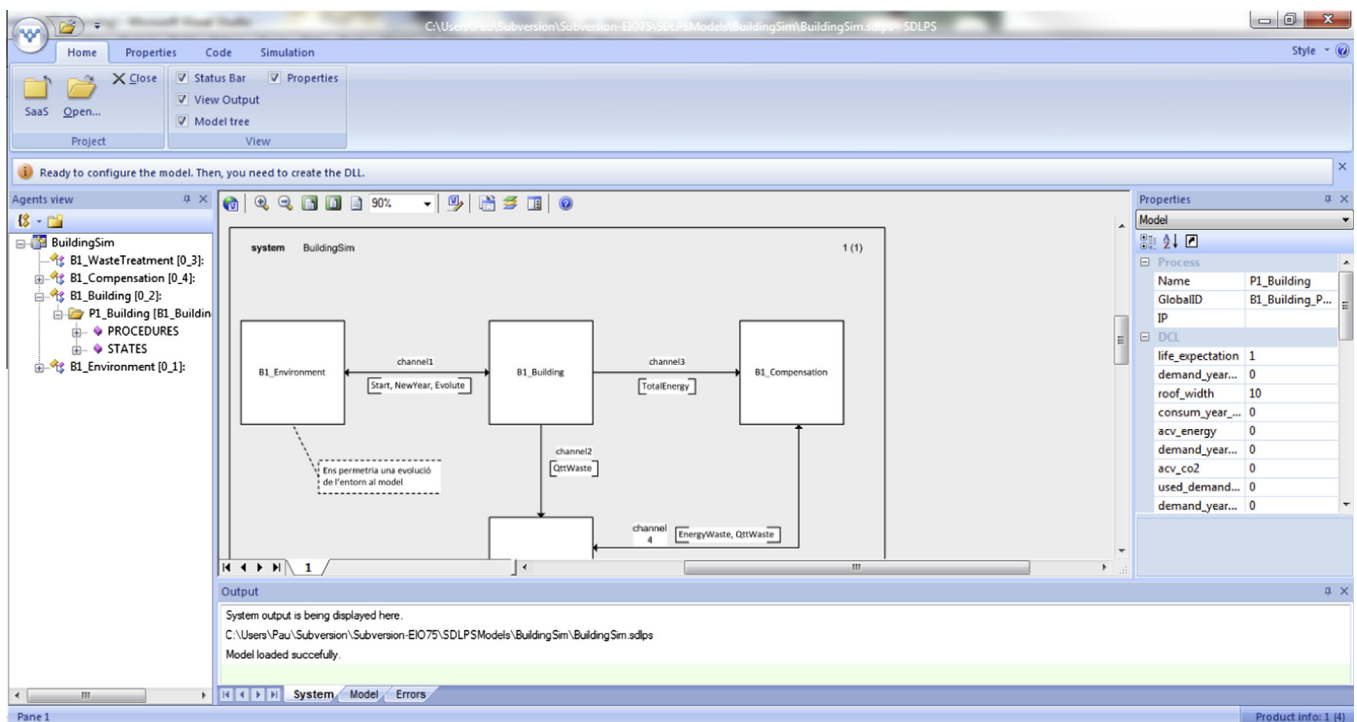


Fig. 10. SDLPS interface with the model.

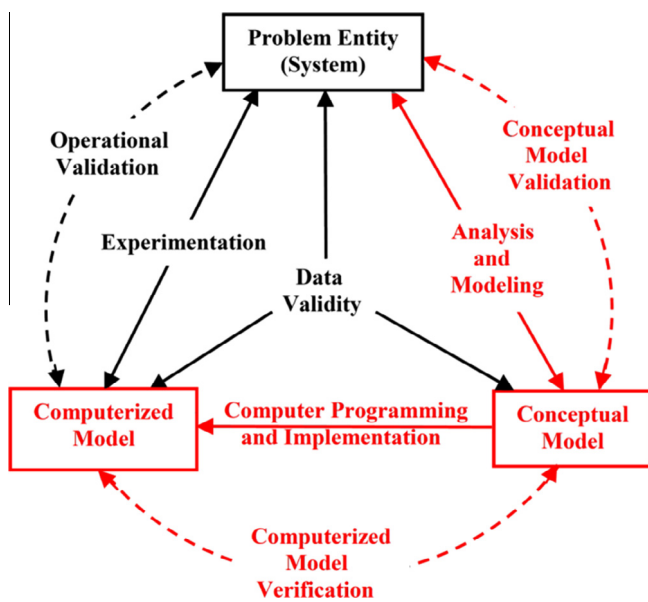


Fig. 11. Simplified version of the Modeling Process [45].

energy+, but in a different way and with the energy+ engine being used directly. This allows us to validate some of the hypotheses made about our model. The complete analysis done here can be reviewed on [53].

With these validations, we ensure that the assumptions of the model related to the structure and the simplification used are correct. This is not enough, however, because we need to validate whether the data that we plan to use in the (e)CO model is correct. To do this, we perform a Face validation [54,55] which is based on the following idea: “Individuals knowledgeable about the system are asked whether the model and/or its behavior are reasonable. For example, is the logic in the conceptual model correct and are the model’s input–output relationships reasonable” [56]. To perform the face validation the formal representation of the model was a valuable tool, since the experts do not need to understand the programming language used to implement the simulator. Based on the unambiguous and complete graphical representation of the model, the experts on the system can validate the model, assuring that the relations and the communications existing between the different model elements are correct and coherent with his understanding of the system. Also the experts analyze the outputs of the model understanding his coherence based on his knowledge of the system.

## 7. Experimentation

The experimental variable that we take into account in this simulation of the (e)CO prototype is the insulation thickness. To conduct the experiment, we define a factorial  $2^K$  experiment. The response value is the energy demand, and we are focused on minimizing this value. The factors we can modify are the different walls of the building, and the levels are two insulation values we can define for each one of the walls. For each of the different walls that exist in the building we categorize them in tree modules (north, south and east), see Fig. 13.

The structure of the experimental design is summarized in the next table. The initial value is 0.212, and the new value (improved insulation) is 0.174.

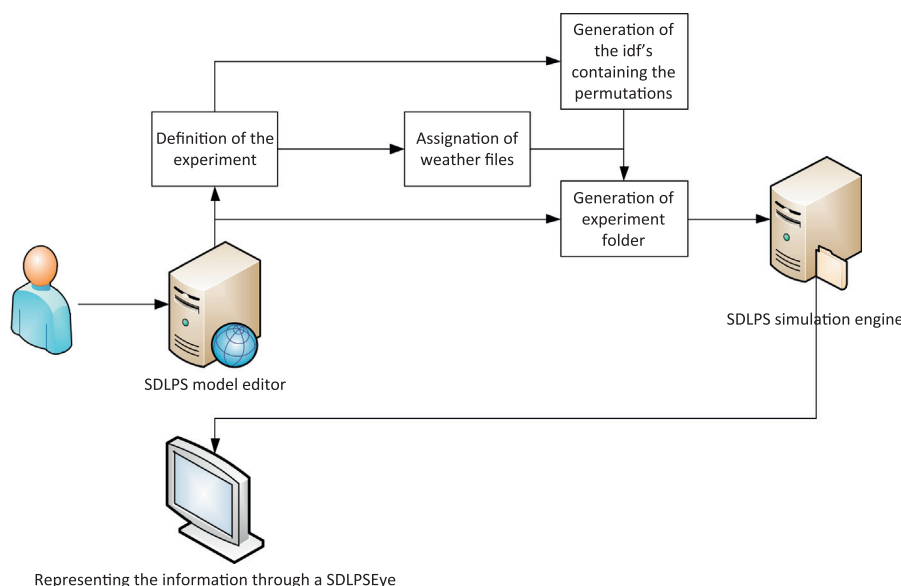


Fig. 12. Generation of scenarios in SDLPS.

$U$ (W/m <sup>2</sup> K)	North MODULE				South MODULE				East MODULE			
	North wall	West wall	South wall	East wall	North wall	West wall	South wall	East wall	North wall	West wall	South wall	East wall
s1	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212
s2	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.174
s3	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.174	0.212
s4	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.212	0.174	0.212	0.212
...	...	...	...	...	...	...	...	...	...	...	...	...
S4096	0.174	0.174	0.174	0.174	0.174	0.174	0.174	0.174	0.174	0.174	0.174	0.174

Each of the different experiments requires approximately 2 min to be completed on a quad core i5 computer, and because we have 4096 experiments, we need approximately 6 days when using a single computer.

SDLPS helps us by facilitating the implementation of a very simple distributed scenario. In our approach, we split all of the experiments into different submodels that can be executed on different machines. Each one of these machines is connected to a single SDLPSEye that receives all the information from all the engines. The instruction that makes it possible to receive this information is the Report procedure call. This procedure belongs to the SDLPS API.

## 8. Results

The main output of the model is the Natural gas consumption (kW/h). The main problem in this model is the amount of data

we obtain from the simulator. To analyze all of this data, because a  $2^k$  factorial design was chosen, we use the Yates algorithm to detect the interactions of the different factors and the main factors of the model.

In Fig. 14, a diagram that shows the Natural gas consumption (kW/h) for each of the different simulations is shown. The pattern presented by the model clearly shows a regular structure in the emissions. To execute the model we divide the different scenarios in four main blocks (scenario 1–1000, 1001–2000 and so on). This pattern is represented in the figure because in the x axis is shown the identifier of the experiment.

In Fig. 15, the relation between the cost and consumption for each of the 4096 simulations is shown. It is possible to detect a set of scenarios that we can consider as optimal or near optimal solutions, which relates a low consumption with low cost.

As it could be difficult to assure what is the best alternative, taking into account the inherent variability of a simulation solution, it is need to use a methodology to analyze the data. For that we use

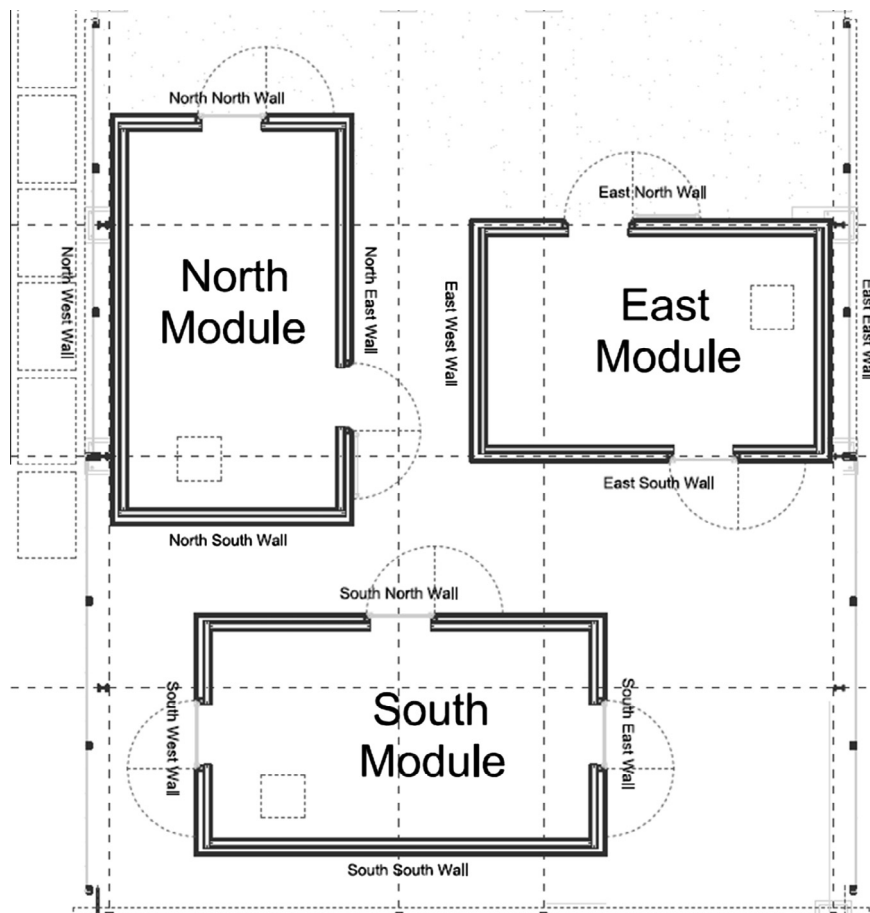
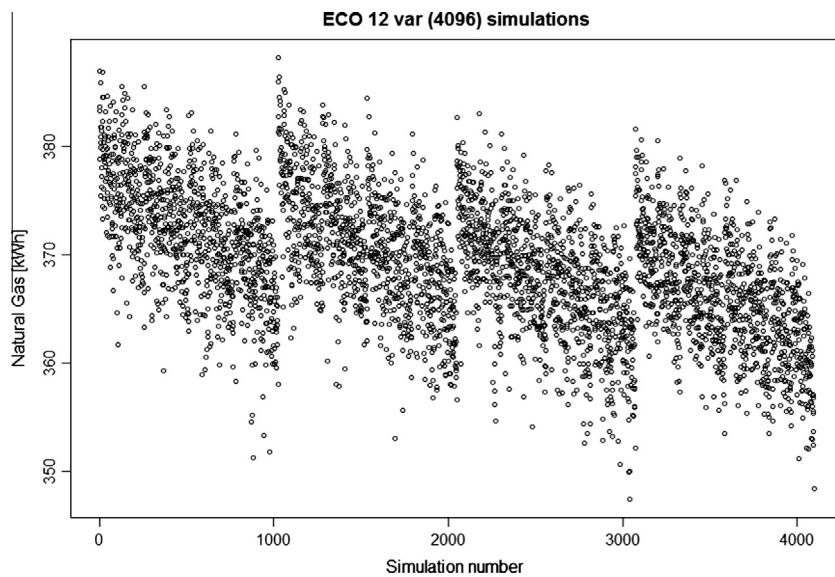


Fig. 13. Figure of the different walls analyzed.



**Fig. 14.** Natural gas (kW/h) emissions for each simulation. On the X axis is represented the identifier of each simulation scenario (1, 2, and so on until 4096) on the Y axis is represented the amount of kW/h (of natural gas) used in the building (less is better).

the Yates algorithm, the mathematical details of the Yates analysis are given in chapter 10 of [57]. The main purpose is to identify the factors (that lead to the different experimental alternatives) that have the most effect on the answer.

These two graphics are obtained automatically from the model execution thanks to the graphical capabilities of the tool and his integration with R language.

Thanks to Yates algorithm we obtain the effects of the factors of the model and have obtained a clear picture of these effects, which are represented in Table 4.

Now, we can clearly analyze which parameters are the most influential regarding the potential for energy savings and the walls we must pay attention. As you can see in Table 4, the effects that maximizes the energy demand savings are those who have bigger values on Effects column, hence depending on the amount of resources we have is interesting to invest first on North Wall of North module, then on South Wall of North module, and then in East Wall of East module, and so on.

## 9. Concluding remarks

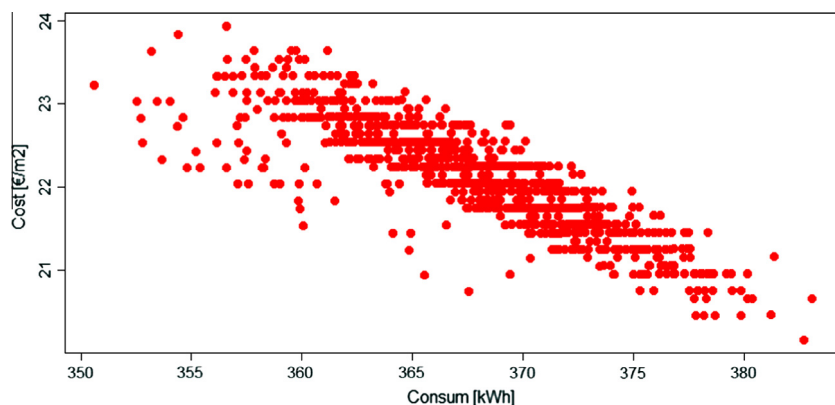
This work presents a novel approach that uses co-simulation to analyze the energy demands of a building. The complete, graphical

and unambiguous definition of the model thanks to a formal complete and standard language such as SDL, simplifies the interaction and the participation of all the actors involved in the development of the model. Nonexperts in simulation can understand the dynamic behavior of the model as a result of the graphical nature of SDL. The team can evaluate the model and propose

**Table 4**

Effects of the factors in our model. With the effects we can detect what are the more influential factors to be taken into account in order to obtain the optimal solution.

Module	Wall	Effects	Mean
East	East wall	4.35	3.34
	South wall	3.09	
	West wall	3.26	
	North wall	2.66	
South	East wall	2.23	2.54
	South wall	2.16	
	West wall	3.44	
	North wall	2.35	
North	East wall	2.74	3.46
	South wall	4.65	
	West wall	1.60	
	North wall	4.86	



**Fig. 15.** Relation between cost and consumption for the 4096 experiments. On the X axis is represented the consumption of gas for each scenario, on the Y axis the cost of each scenario (€/m<sup>2</sup>). The best alternatives are those who minimizes consume and cost (each dot represents one of the 4096 experiments performed).



modifications using common tools such as Microsoft Visio®. This approach simplifies the involvement of the specialists in the model definition, a key element to success in a complex simulation project [54]. The formal representation of the model used to perform its validation, allows to detect errors of interpretation before any implementation.

This is an structured model that allows to increase, thanks to the modularity of the approach, the complexity of the model, and interact with other models and systems (like those defined formally with DEVS or Petri Nets as an example).

The model using open source calculus engines, allows the optimization of the analyzed system following the European normative CEN/TC 350 (UNE-EN 15643-2, UNE-EN 15643-3, UNE-EN 15643-4).

Also, the automatic mechanisms existing to execute the model, without the need of implement it, simplifies the verification process. Co-simulation approach allows the combination, in our case using SDLPS, of several sub-models and calculus engines that simplifies the definition of the dynamic behavior of the building. Specifically, the use of energy+ makes it possible to precisely calculate the energy demand as is shown on [53].

From the point of view of the infrastructure used to implement the model, it is interesting to remark that it could be executed on a cloud infrastructure, allowing to interact with the model with a common web browser that shows the basic parameterization elements of the model. Also thanks to the parallel structure of SDL it is easy to parallelize the execution of the model on SDLPS.

Regarding the results obtained on the case study analyzed, it is notable that not all of the walls of the building react the same way to the modification of the parameters. This clearly implies that it is not necessary to invest in all of the constructive elements of the building to achieve an optimal solution. We detect where to invest, the more interesting walls to consider and the actuators that must be done, in order to reduce the energy consumption; the use of a  $2^K$  factorial design with the help of a posterior Yates algorithm, helps in an easy interpretation of the data.

As a result of using a formal language like SDL to define the proposed integral sustainability model, it will be possible in the future to optimize the studied model throughout the four phases of the construction process of a building by taking into consideration the environmental, economic and social impacts. In addition, it is possible to define the process needed to compensate for the demanded and the consumed energy.

The methodology and the tools proposed are currently used successfully in a real project to improve the behavior of a new architectural design comparing several alternatives.

## Acknowledgements

We appreciate the support of the inLab FIB of the Polytechnics University of Catalonia.

We acknowledge the LivingLab Low3 project directed by Torsten Maseck.

We appreciate the (e)CO team—equilibrium through cooperation—the group that designed the prototype building for the Solar Decathlon 2012 competition.

## References

- [1] European directive. 2010/31/CE. UE; 2010.
- [2] International Energy Agency. International Energy Agency; 2012. <<http://www.iea.org>> [accessed 31.05.12].
- [3] Salom J, Widén J, Candanedo J, Sartori I, Voss K, Marszal A. Understanding net zero energy buildings: evaluation of load matching and grid interaction indicators. In: Proceedings building simulation, 12th conference of international building performance simulation association, Sydney, 14–16 November, 2011.
- [4] Sartori I, Napolitano A, Voss K. Net zero energy buildings: a consistent definition framework. *Energy Build* 2012;48:220–32.
- [5] Palonen M, Hasan A, Siren K. A genetic algorithm for optimization of building envelope and HVAC system parameters. In: Proceedings of the eleventh international IBPSA conference, Glasgow; 2009.
- [6] Salom J. Els edificis de consum d'energia quasi zero. *Cultura Energètica* 2012;179:14–7.
- [7] BPIE – Buildings Performance Institute Europe. Principles for nearly zero-energy buildings. Paving the way for effective implementation of policy requirements; 2011.
- [8] Salom J, Cubí E, Sánchez A. Edificios de balance energetico cero: Buscando una definicion concreta para un objetivo amplio. *CIC Arquitect Construcc* 2011;485:54–5.
- [9] Salom J, Cubí E, Sartori I. Edificio de energía cero: definiciones e interacción con las redes energéticas. In: Libro de Comunicaciones I Congreso EECN, Madrid; 2012.
- [10] McDonough W, Braungart M. *Cradle to Cradle: remaking the way we make things*. 1 ed. North Point Press; 2002. p. 208.
- [11] Arantes L, Olivier B, Pascal R, Daniel Q. A simple method to consider energy balance in the architectural design of residential buildings, Boston; 2011.
- [12] Andreas Sattrup P, Strømmand-Andersen J. A methodological study of environmental simulation in architecture and engineering. Integrating daylight and thermal performance across the urban and building scales. In: 2011 Proceedings of the symposium on simulation for architecture and urban design, Boston; 2011.
- [13] Sodja A, Zupancic B. Modelling thermal processes in buildings using an object-oriented approach and Modelica. *Simul Model Pract Theor* 2009;17:1143–59.
- [14] Counsell JM, Khalid YA, Brindley J. Controllability of buildings: a multi-input multi-output stability assessment method for buildings with slow acting heating systems. *Simul Model Pract Theor* 2011;19:1185–200.
- [15] Liu B, Duan S. Energy efficiency evaluation of building integrated photovoltaic systems with different power configurations. *Simul Model Pract Theor* 2012;29:93–108.
- [16] Bruno M, Henderson K, Min Kim H. Multi-objective optimization in urban design. In: 2011 Proceedings of the symposium on simulation for architecture and urban design, Boston; 2011.
- [17] U.S. Department of Energy. “GenOpt”; 2011. <<http://www.simulationresearch.lbl.gov/GO/index.html>> [accessed 31.05.12].
- [18] National Renewable Energy Laboratory. “Opt-E-Plus”, U.S. Department of Energy, October 2011. <[http://www.nrel.gov/buildings/energy\\_analysis.html](http://www.nrel.gov/buildings/energy_analysis.html)>.
- [19] NREL. “BEopt”; 2012. <<http://www.beopt.nrel.gov/>> [accessed 31.05.12].
- [20] Adams B, Bohnhoff W, Dalbey K, Eddy J, Eldred M, Gay D, et al. DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.0 User's Manual; 2009.
- [21] Zeigler B, Praehofer H, Kim D. *Theory of modeling and simulation*. Academic Press; 2000.
- [22] ITU-T. Specification and Description Language (SDL). International Telecommunication Union; 2012. <<http://www.itu.int/ITU-T/studygroups/com17/languages/index.html>> [accessed 11.12].
- [23] Recalde L, Teruel E, Silva E. Autonomous continuous P/T systems. Application and theory of Petri nets. *Lect Note Comput Sci* 1999;107–26.
- [24] Sayed Ahmed A, Wainer G, Mahmoud S. Integrating building information modeling & Cell-DEVS simulation; 2010.
- [25] Trcka M, Hensen JL, Wetter M. Co-simulation for performance prediction of integrated building and HVAC systems – an analysis of solution characteristics using a two-body system. *Simul Modell Pract Theor* 2010;18:957–70.
- [26] Gotzhein R, Krämer M, Lothar L, Chamaken A. Energy-aware system design with SDL. In: *SDL 2009: design for motes and mobiles*. Bochum: Springer; 2009.
- [27] Griffith D, Crawley B, Hand JW, Kummert M, Brent T. Contrasting the capabilities of building energy performance simulation programs; 2005.
- [28] U.S. Department of Energy. EnergyPlus; 2013. <<http://www.apps1.eere.energy.gov/buildings/energyplus/>> [accessed 25.06.13].
- [29] U.S. Department of Energy. EnergyPlus Energy Simulation Software; 2012. <[http://www.apps1.eere.energy.gov/buildings/energyplus/energyplus\\_testing.cfm](http://www.apps1.eere.energy.gov/buildings/energyplus/energyplus_testing.cfm)>.
- [30] Hirsch JJ. eQuest; 2010. <<http://www.doe2.com/equest/index.html>> [accessed 31.05.12].
- [31] IEC International Engineering Consortium. SDL Tutorial; 2000. <<http://www.iec.org/online/tutorials/sdl/>> [accessed 01.09].
- [32] Fonseca P. Simulation hypotheses. In: *SIMUL 2011*, the third international conference on advances in system simulation, Barcelona; 2011.
- [33] Comité técnico AEN/CTN 198 sostenibilidad en la construcción. Evaluación de la sostenibilidad en los edificios. Parte 4: marco para la evaluación del comportamiento económico, Madrid: AENOR; 2012.
- [34] Comité técnico AEN/CTN 198 sostenibilidad en la construcción. Evaluación de la sostenibilidad en los edificios. Parte 2: marco para la evaluación del comportamiento ambiental, Madrid: AENOR; 2012.
- [35] Comité técnico AEN/CTN 198 sostenibilidad en la construcción. Evaluación de la sostenibilidad en los edificios. Parte 3: marco para la evaluación del comportamiento social, Madrid: AENOR; 2012.
- [36] (e)CO team. (e)co model, Solar Decathlon Prototip; 2012. <[http://www.solardecathlon.upc.edu/2012/e\\_dise%C3%B1o.html](http://www.solardecathlon.upc.edu/2012/e_dise%C3%B1o.html)>.
- [37] Low3 team. Low3 – Prototip model SolarDecathlon 2010; 2010. <<http://www.low3.upc.edu/INTRO.html>>.

- [38] inLab FIB. Building simulation, 15 04, 2012. <<http://www.inlab.fib.upc.edu/ca/simulacio-energetica-dedificis>> [accessed 01.05.12].
- [39] Fonseca P. SDL distributed simulator. In: Winter simulation conference 2008, Miami; 2008.
- [40] Fonseca P, Casanovas J. Towards a SDL-DEVS simulator. In: The third international conference on advances in system simulation, Barcelona; 2011.
- [41] Fonseca P, Colls M, Casanovas J. Towards a representation of environmental models using specification and description language. In: Proceedings on the international joint conference on knowledge discovery, knowledge engineering and knowledge management; 2010.
- [42] Fonseca P, Colls M, Casanovas J. Representing Fibonacci function through cellular automata using specification and description language, Ottawa; 2010.
- [43] Fonseca P. Using Specification and Description Language to define and implement discrete simulation models. In: Proceedings of the 2010 summer simulation multiconference; 2010.
- [44] Balci O. Golden rules of verification, validation, testing, and certification of modeling and simulation applications. *SCS M&S Magz* 2010;4:1–7.
- [45] Sargent RG. Verification and validation of simulation models. In: Proceedings of the 2007 winter simulation conference, Washington; 2007.
- [46] PragmaDev SARL; 2012. <<http://www.pragmadev.com/product/codeGeneration.html>>.
- [47] CINDERELLA SOFTWARE. Cinderella SDL; 2007. <<http://www.cinderella.dk>> [accessed 31.03.09].
- [48] Fischer J, Neumann T, Olsen A. SDL code generation for open systems. *Lect Note Comput Sci* 2005;3530:313–22.
- [49] Fonseca P. Enhancing SDLPS with co-simulation. In: Proceedings of the 2012 winter simulation conference, Berlin; 2012.
- [50] Balci O. Verification, validation and accreditation of simulation models. In: Proceedings of the 1997 winter simulation conference; 1997.
- [51] Sargent RG. Verification, validation and accreditation of simulation models. In: Proceedings of the 2000 winter simulation conference; 2000.
- [52] Design Builder Software. DesignBuilder; 2005. <<http://www.designbuilder.co.uk/>>.
- [53] Fonseca P, Fonseca A, Garrido N, Colls M, Casanovas J. Building optimisim, optimizador NZEB. In: Greencities y Sostenibilidad: 3er Salón de la Eficiencia Energética y Sostenibilidad en Edificación y Espacios Urbanos, Málaga; 2012.
- [54] Law AM. How to build valid and credible simulation models. In: Proceedings of the 2005 winter simulation conference; 2005.
- [55] Weiner IB, Craighead WE. Face validity, the Corsini encyclopedia of psychology; 2010. p. 637–8.
- [56] Sargent RG. Verification and validation of simulation models. In: Proceedings of the 2009 winter simulation conference; 2009.
- [57] Box GEP, Hunter JS, Hunter WG. *Statistics for experimenters: an introduction to design, data analysis, and model building*. 1 ed. Wiley-Interscience; 1978. p. 664.