SUSTAINABILITY-OPEN: WHY EVERY BUILDING WILL BE SUSTAINABLE IN THE FUTURE

Abstract
In this paper the initiative “sustainability-open” will be introduced. The aim of the initiative is to take away one of the reasons why buildings and other objects in the built environment are often not designed in a sustainable manner: the availability of clear and insightful software to analyse and assess the sustainable performance of the design.

Most software applications designed for this purpose do not reveal their internal and implicit method to designers and cannot be easily adapted to the purposes of the designer or engineer. “sustainability-open” consists of an open-source framework on which both open-source and close-source components can be build which contain the specific design, analysis and assessment knowledge for these purposes. The development consists of a framework and components. The open-source framework and the components developed by the research team are available free of charge, which means that cost is no longer an issue to design in a sustainable manner. The initiative is open for people and companies to contribute and extend by development of their own, optionally closed-source components. The framework development as well as a set of components, which are based on the research projects performed at the BEMNext Lab of Delft University of Technology, are managed by the research team. However, this should not stop anybody to develop and contribute new components to initiative.

The current framework includes concepts for automated ‘designers’, ‘analysis methods’ and ‘assessment methods’. Automated designers perform automated tasks or methods to produce a digital design or contribute to an existing, static design. ‘Designers’ can be used to encode some of the knowledge the designing engineers have. Analysis components perform some type of analysis, such as a thermal analysis, to produce analysis results, which can be used in further analysis or assessments. Assessment components produce the assessment, such as the total amount of energy used in the building, the operating energy or the embodied energy.

The framework currently integrates with Grasshopper (Davidson 2013) and development is taking place to integrate with a variety of BIM software (Eastman 1999) applications.

The paper will present the initiative, the methodology behind the development, the framework and different components, and an outlook towards further research and developments, which will be implemented in the (near) future.

Introduction
In this paper the initiative “sustainability-open” will be introduced. The aim of the initiative is to take away one of the potential reasons why buildings and other structures and objects in the built environment are often not designed in a sustainable manner: the availability of clear and insightful knowledge how to design, analyse and assess the sustainable performance of the design. The reasoning is simple: if it is not easy to measure how sustainable a design is or will be, it is hard to produce sustainable design alternatives, influence the sustainable performance of concepts, the choice for sustainable design alternatives, building components, optimise for sustainable performance, etc. The “sustainability-open” initiative aims to solve this issue by providing software developers, designers and engineers to encode this knowledge in software that can help the designer to design, analyse and assess the sustainable performance of an object in the built environment. Where this process can be automated the initiative allows for encoding this automation as well.
Sustainable design

The initiative also recognizes that top-down definition of a ‘universal truth’ for sustainable design is very hard to define as sustainable performance can be measured in many different ways and many different databases of materials and their performance exist as well as many different evaluation methods. Also many different software applications exist which encode these methods, but largely keep their knowledge implicit and closed-source.

“sustainability-open” does not have preference for a single performance measure, but allows for many different measures for sustainability. One can think of measures, like

- Material use: the quantity of the different materials used in the object, e.g. measured by weight or volume.
- Material depletion: the quantity of scarce materials used in the object, again measured by weight or volume. Several important construction materials are becoming more and more scare.
- Embodied energy: the amount of embodied energy in the materials from production, transformation and transportation of the materials, e.g. measured in (kilo/Mega)Joules.
- Energy usage: the total amount of energy that the object uses, e.g. the operational energy.
- Pollution: the amount of pollution that is caused by the object.
- Radiation: the amount of radiation (potentially) being released by the object.
- Toxicity: the amount of toxic materials used.
- Emissions: the amount of emissions released from the object.

A number of these measures can be measured as an initial amount used or over the total life cycle of the object. Not all measures have a negative impact of the sustainable impact of an object:

- Recycling: the amount of material from the object that can be recycled.
- Up-cycling: the amount of material from the object that potentially can be up-cycled (transformed in a higher quality material, e.g. low grade steel that can be up-cycled to a higher grade steel).
- Energy production: the amount of energy produced by the object, e.g. by making use of PV cells.
- Energy storage: the amount of energy that can be temporarily stored by the object to reduce energy consumption over a larger time scale.
- Adaptability: the fact if the object can be used for multiple purposes and therefore might extends its life cycle by exploiting a second or third functionality after full exploitation of the first use.

The initiative allows the designer to make up their own mind based on their professional experience what the best methodology is to apply the specific design case. The open-source nature of the initiative allows for many performance measurement techniques, many types of assessment, combinations of assessments, many different products and building components, etc.

Open source

Most software applications designed for the purpose of analysing or assessing sustainability do not reveal their internal methodology and/or methods to designers and cannot be easily adapted to the purposes of the designer or engineer. “sustainability-open” consists of an open-source framework on which both open-source and close-source components can be build. This means that designers and engineers potentially can have full insight in the workings of the framework and in case they choose to use the open-source components (for further explanation on the distinction between the framework versus components, see below) full insight in the components as well. On the other hand in case for example that a company has a proprietary in-house design method, it can still use the framework by encoding their knowledge in closed-source components and not sharing these with the public. The implications of insight in methodology and methods is that designers can have a better understanding of how the methods work, have the opportunity to learn where the ‘sweet spots’ are, are able to verify the correctness of the data and methods used and can extend the software and easily customise it for project-specific needs. Of course, it needs to be noted that in order to gain this insight the designers will need some software development knowledge to
directly understand the source code. On the other hand, at least the “sustainability-open” initiative gives insight in contrast to purely close-sourced initiatives. The open nature of the methodologies uses is also important for science and scientific validation. Any scientist is able to download the source code to verify the correctness of the methods used.

Another important aspect is the fact that the open-source framework and the components developed by the research team are available free-of-charge, which means that cost (of software) is no longer an issue to design in a sustainable manner.

The initiative is open for people and companies to contribute and extend. The research team and contributors manage the framework development. The research team is also developing a set of components, which are based on the research projects performed at the BEMNext Lab of Delft University of Technology and a variety of contributors and collaborators. However, everybody is stimulated to develop and contribute new components to initiative. All code is hosted on Github (Github Inc 2013) and can be forked to develop derivatives.

In the remaining parts of the paper the “sustainability-open” initiative, the methodology behind the development, the concepts of the framework and different components, some details on implementation and an outlook towards further research and developments which will implemented in the (near) future will be presented.

Methodology and high-level software architecture
The initiative consists of two parts: the framework and components developed based on the framework. The framework provides all base functionality so that components can be built on top of this framework to contain the encoded knowledge. The framework contains mainly the core data structure in the form of abstract classes from which components can be inherited. Also some common often-recurring helper classes have been made part of the framework to reduce unnecessary redevelopment. As the initiative is in constant flux as an open-source project all explanation below is based on the current framework (version 0.0.1) unless the explanation points to higher versions or concepts to be implemented in future versions.

Figure 1: Simple Grasshopper (Davidson 2013) model with two Designer components (“SimpleDesigner” implementation), one Analysis component (“QTO analysis implementation”), one Assessment component (“AssessmentExample” implementation) and two native Grasshopper components to show some textual output of the components.
Framework

Fig. 2 shows a piece of the software architecture of the framework. The current framework includes concepts for automated designers, analysis methods and assessment methods. Automated designers perform automated tasks or methods to produce a digital design or contribute to an existing, static design. Designers can be used to encode some of the knowledge designers have. An example would be a designer, which automatically produces a steel column-beam frame structure, which is one of the designers developed by the research team. A limitation in the current framework is that designers will need to be able to produce a design in a single automation step (it must be encoded in a single overridden method on the designer class).

Analysis components perform some type of analysis, such as a thermal analysis, to produce analysis results, which can be used in further analysis or assessments. Again the limitation of this analysis is that it should be able to complete in a single automation step.

Assessment components produce the assessment, such as the total amount of energy used in the building, the operating energy or the embodied energy.

Besides these three main components, the framework provides a number of classes to build objects in the form of so-called “physical objects” which contain a material quantity.

![UML class diagram](image)

Figure 2: UML class diagram of the “sustainability-open” framework depicting the three base classes: SODesigner, SOAnalysis and SOAssessment.
Figure 3: Detailed view of the UML class diagram of the three base classes. Important to note are the “RunDesigner”, “RunAnalysis” and “RunAssessment” methods which should overridden in case of a custom implementation.

The framework currently also adds a quantity take-off analysis, which simply adds up the quantity per material for all objects produced by the automated designers, further discussed below.

Components

Components are collections of objects that are derived from one of the base classes in the framework. These components are intended to contain the encoded knowledge in either open-source or closed-source form.

As stated, the current framework provides a single component: the QTO component, which is derived from the SOAnalysis class. This component performs a quantity take-off based on the materials in the design. Fig.4 shows a class diagram of the QTOAnalysis class and its relationship to the SOMaterialQuantity component which is used to register the material quantities. Because the class inherits from the SOAnalysis class, it contains a set of Designers as input which are used to perform the quantity take-off on. As can be seen in Fig.2 each Designer contains a number of SOPhysicalObjects which contain material quantities.

Other examples of components under development by the research team are a component to assess the total amount of embodied energy based on the materials used (calculated by the QTO), a component to assess the total operating energy used by a building used over its life cycle and a measure for adaptability of the building.

Plans exist to build components in line with the Bath database “Inventory of Carbon and Energy” (Hammond 2008).
Figure 4: UML class diagram, which shows the functionality of the QTOAnalysis class, which is used to perform quantity take-off analysis on the object.

**Implementation**

The framework has been implemented on the .NET framework (Microsoft 2013) and C# as programming language (Perry 2006). Test-driven development has been used through NUnit as a unit testing framework (Nunit.org 2013) to manage the source code on a detailed level.

The framework currently integrates directly with the parametric modelling system Grasshopper (Davidson 2013) as uses its parametric functionality in order to visualise the design, analysis and assessment logic, to drive the update cycle and to visual potential design results (e.g. the object itself or graphs). Development is taking place to integrate with other parametric systems and a variety of Building Information Modelling (BIM) software (Eastman 1999) applications to maximise its usability in practice. In order to achieve this the framework will be further abstracted so that it is able to run independently from Grasshopper or any other software. These software applications will be used only visualise and facilitate the modelling process.

The current architecture of the framework already allows the framework to run independently, except for the fact that little infrastructure exists to add custom code without recompiling the framework and components.
Licensing

The “sustainability-open” initiative is being released under several open-source licenses in order to facilitate maximum use in practice. Companies often find so-called ‘copyleft’ licenses, software-licenses that require derivatives of the original source code to be publically shared back, limiting as this might expose what they feel is their business advantage. The framework is released under an Apache 2.0 license, which does not impose any copyleft restrictions. Therefore, companies and individuals can develop their own closed-source functionality without the need to share (but still are strongly encouraged to share as this will make everybody’s life better in the long run). In contrast to the framework impose the components developed by the research team copyleft restrictions through their licensing. The reasoning behind this is that work, which is the result of a scientific research benefits from peer review, scientific verification and sharing of research results with other groups.

Next version of the framework

Currently, the second version of the sustainability-open framework is being implemented. This next version of the framework will introduce a number of improvements compared to the previous version. First, more possible relationships between the different components in the framework are supported. Where in the initial framework only Designer-to-Analysis and Analysis-to-Assessment relationships where supported, which limited the models in the framework to a three-layer structure, in the next version also Designer-to-Designer, Analysis-to-Analysis, Assessment-to-Assessment and Designer-to-Assessment relationships have been introduced. This allows for multi-layer structure, which fits much closer to realistic and logical modelling of design problems.

In the next version the concept of a ‘Design’ is introduced. This object describes a design solution by specifying the ‘Components’ it consists of (note the difference between these design components and the components which are being plugged into the framework). These components can consist of other components and/or physical objects. A ‘Physical Object’ is defined as a single-material design component which a fixed quantity. This ‘Quantity’ is measured by a ‘Unit’ and a ‘Value’. So for example a Design can be a chair, which consists of a number of components: the seat, the legs and the back. The seat might consist of a cushion and a wooden seat. The wooden seat ‘physical object’ might be of material ‘wood’ and consists of 0.25 m$^2$ seating area and 0.03 m$^3$ wood. Note that on the physical object in this example two quantities have been defined. Materials can have various ‘Properties’, each consisting of a ‘Description’ of the property, the ‘Unit’ and the ‘Value’. These properties can be used for a variety of uses, such as analysis input.

This is an important development towards Building Information Modelling, which documents designs in its data structure. Together with the introduction of the ‘Design’ concept also manual input will aid the connection to BIM software as design do not necessary have to originate from automated designer components, but can be the result of a pre-modelled 3D model documented in a BIM software application. Potentially, also a combination of automated and manual design can be used by making use of the new multi-layer capabilities.

Also, a component library is envisioned so that pre-developed components can be easily encoded and shared. This also aids the connection to BIM software.

Discussion

“sustainability-open” helps to improve our world in the view of the author by providing open-source knowledge that can be used to design better and more sustainable buildings and structures. The open-source nature of the project might also be one of the weaknesses, which will need to be evaluated over time. For example the question if people will actually contribute to the projects and start their own projects cannot be easily answered. Many examples of both successful and unsuccessful open-source communities and products exist. Another question, which is difficult to foresee is if people are willing to encode their knowledge.
The current framework and components have a number of limitations, which largely have been discussed, in the section on the next version of the framework. Additionally, an important limitation is the dependency on Grasshopper, which will be removed, in one of the future versions. This will open the road to integrations with other software applications.

**Conclusions**

This paper has presented the “sustainability-open” initiative, which provides open-source software tooling for sustainable design in the built environment. The paper has presented the methodology, core concepts and implementation. Furthermore, the paper has explored limitations and further ideas for the future expansion of the framework and new components to be developed.

**Acknowledgements**

The author would like to acknowledge Tata steel, the Dutch Steel Institute (Bouwen met Staal), Delft University of Technology and Arup for their support for and involvement in the initial research project.

**References**


