D6.6

Report on improved data integration platform

Deliverable Report: 6.6Final version

Streamer - Optimised design methodologies for energy-efficient buildings integrated in the neighbourhood energy systems.
The Streamer project is co-financed by the European Commission under the seventh research framework programme FP7.EeB.NMP.2013-5 ; GA No. 608739)

D6.6

Report on improved data integration platform / open server for data and tools

Main author Léon van Berlo (TNO)

Co-authors Pim van den Helm (TNO),

Dissemination Public

Document history

Version	Date	Status	Produced by	Comments
V0.1	2017-03-01	First Draft	Léon van Berlo	D6.8 final is used for this
V0.2	2017-07-18	Final draft	Pim van den Helm	
	2017-02-27	Reviewed	Nisbet; Cartwright; Bruinenberg; Haefele; Benner; Hempel	
V1	2017-08-10	Final	Pim van den Helm; Léon van Berlo	Sent for review
V2	2017-08-18	Final	Pim van den Helm	Comments review processed
V3	2017-08-21	Approved	Freek Bomhof	

Colophon

The Streamer project is co-financed by the European Commission under the seventh research framework programme with contract No.: 608739 - FP7-2013-NMP-ENV-EeB. The information in this publication does not necessarily represent the view of the European Commission. The European Commission shall not in any way be liable or responsible for the use of any such knowledge, information or data, or of the consequences thereof.

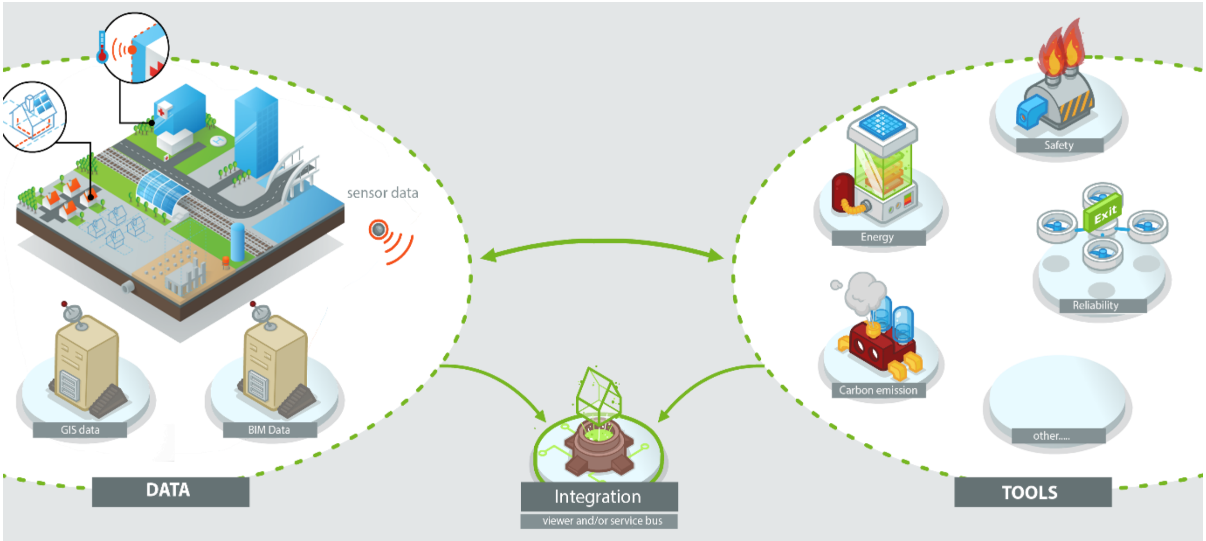
Publishable executive summary

Within the STREAMER project the challenge is to provide design teams with advanced design tools by improving the open interoperability between Building Information Modelling (BIM) and Geographical Information System (GIS) as to enable model-based prediction of the energy performance.

It is the objective to effectively manage information flow, knowledge integration, communication and decision-making in the participatory semantics-driven design process according to the principles of Integrated Project Delivery [Bektas2016].

In the design phase of a complex project like a healthcare district, the data varies widely. These data are highly distributed and heterogeneous, thereby implying the use of multiple models and resources. This report will define the base for a possible solution on how BIM and GIS-based assessments tools can be coupled with the product lifecycle model (PLM architecture) in order to be straightforward and easy to use [Vial2015].

Given the situation and analysis, we propose a combination of technologies to be used in STREAMER for an improved data and tools integration. The concept is sketched in the following image:



Generally, the STREAMER concept consists of three categories:

- The tools (simulations)
- The data (BIM, GIS coming from authoring tools, and data from sensors)
- The integration platform

To limit the number of one-to-one connections the concept of an ‘Enterprise Service Bus’ (ESB) is introduced as a concrete solution to the ‘Information Logistics’ layer in the PLM architecture. An ESB is implementing a communication system between mutually interacting software applications in a service-oriented architecture (SOA).

Based on previous experiences and first experiments, we recommend the use of WSO2 Enterprise Integrator [WSO2_2017] and Zapier Developers platform [Zapier2017] to open unlock the ‘Information logistics’ layer of the proposed PLM architecture.

A solution to connect the several different entities is with a so called ‘event driven’ approach.

List of acronyms and abbreviations

- API: Application Programming Interface
- BIM: Building Information Model
- CAAD: Computer Aided Architectural Design
- ESB: Enterprise Service Bus
- GIS: Geographic Information System
- IPD: Integrated Project Delivery
- OGC: Open Geospatial Consortium
- PLM: Product Lifecycle Manager
- SOA: Service-Oriented Architecture
- STREAMER: Semantics-driven Design through Geo and Building Information Modelling for Energy-efficient Buildings Integrated in Mixed-use Healthcare Districts
- TNO: Netherlands Organisation for Applied Scientific Research
- WFS: Web Feature Service
- XML: Extensible Markup Language

Definitions

Web Service –. A web service is a service offered by an electronic device to another electronic device, communicating with each other via the World Wide Web [WebService2017]

Microservice - Microservices is a variant of the service-oriented architecture (SOA) architectural style that structures an application as a collection of loosely coupled services. There is no industry consensus yet regarding the properties of microservices, and an official definition is missing as well. [Microservice2017]

BIM Bots - A bot is an (online) system that triggers on an event. It performs a task with some form of "intelligence". A BIM bot can perform an analyses or simulation on your BIM data, it can enrich the model by adding detailed objects, splitting it for specific use, or perform any other task on your data. [BimBots2017]

Contents

1. Introduction

Within the STREAMER project, the challenge is to support design teams with advanced design tools by improving the open interoperability between Building Information Modelling (BIM) and Geographical Information System (GIS) so as to enable model-based prediction of the energy performance.

It is the objective to effectively manage information flow, knowledge integration, communication and decision-making in the participatory semantics-driven design process according to the principles of Integrated Project Delivery.

In the design phase of a complex project like a healthcare district, the data varies widely. The data are highly distributed and heterogeneous, thereby implying the use of multiple models and resources. This report will define the strategy how BIM and GIS-based energy performance assessments tools can be coupled with the product lifecycle models in order to be straightforward and easy to use.

2. Integration of Data

On the topic of integration of data, the most basic integration of data is using a single data format or standard. All data can then be stored in a single location in a single defined format. The choice of the right data model is always difficult.

IFC is the only standardized model for buildings, which claims to cover the complete life cycle [Haefele2015]. Accordingly, the IFC model is very comprehensive and supports nearly 770 entities and nearly 410 property sets (IFC4). With IFC4, the building model can have arbitrary geospatial references and generalized civil engineering (e.g. pavements) and geographical elements (e.g. roads, trees). Usually, not all the geometry is described explicitly. Hence, the importing application needs basic geometrical functions at least. The import and export of IFC is supported by free and commercial tools, but needs more experience than e.g. XML. Currently, IFC is supported by all major CAAD vendors and most of them have already been certified or are in an on-going certification process performed by BuildingSMART International (bSI).

Theoretically, IFC might be able to cover all processes, which are planned within STREAMER. One important reason why this approach does not work in reality, is that when generating a digital representation of buildings, the complete life cycle is not fully considered in an integrative way. There are many actors in many domains with many software applications, which have their own view of the building. All want to have models, which exactly represents the information required and speak their languages. This leads to domain-specific, specialized data models, like gbXML or ILCD. On the other hand, it is not easy to synchronize two major domains, such as the GIS and BIM worlds. The responsible standardization organizations (OGC and bSI) started nearly at the same time (20 years ago) to independently develop standards for their domains. Of course, software applications of the GIS world most likely support OGC standards, while BIM applications provide better support for bSI standards. For designing buildings, it is essential to consider the neighbourhood to obtain a holistic view on the project.

STREAMER Deliverable 6.5 gives an overview of "*Advance Mapping Structures and Standards*". In Chapter 2 modelling languages, data models, data formats, and standardised web services, which might be relevant to STREAMER, are introduced. The D6.5 report also concludes that STREAMER has to consider more than one standard. Depending on data availability and project phase, different data models might be deployed. STREAMER has to consider using different data models in a combined manner in order to realize the intended workflow.

If a process requires more than one data model, for instance, when considering a detailed building model in its neighbourhood, a so-called hybrid approach is required. In this approach, data are kept in the original models and:

- enriched by other information sources,
- converted into the requested target models,
- integrated into a single software application, or
- linked, by the corresponding software tools, if necessary.

In order to bridge the gap between GIS and BIM worlds, models can be converted, federated or linked. A typical example is to generalize a detailed IFC model to a certain LoD and save it as a CityGML model. If an application covers more than one domain and is able to import different models, further analysis and simulations can be performed on the integrated model.

As IFC is the most mature open data model for BIM which covers the whole life cycle of the building and which can also be used in the construction process, it is recommended for use as a basis for all STREAMER processes. In general, this means that IFC is used for designing the new or refurbished building. Depending on the deployed software application and on additionally required information, models can be enriched, converted, or linked.

Data redundancy is a term to define a situation when data in a database has no purpose for existence. Data redundancy leads to data anomalies and corruption and generally should be avoided by design [Peter Rob; Carlos Coronel (2009). Database systems: design, implementation, and management. Cengage Learning. p. 88. ISBN 978-1-4239-0201-0. Retrieved 22 January 2011.]
By enriching, linking and converting models in STREAMER, there is a risk for data redundancy. If different data models describe the same real world object (e.g. a building), it is likely that similar or even the same information is stored multiple times. However, this does not mean that the information is redundant within the process, since the duplicate data is likely to have a specific purpose in the process, and is an integral part of the (update and maintain) dataflow.

The management of different data models including data linking, web services, and preserving data consistency, is difficult to perform manually. Therefore, the next chapter will highlight options for the integration of tools.

3. Integration of tools

Tools can be offline (desktop applications) or online (connected to the internet). Integration of desktop tools is not a topic related to the STREAMER project, so this chapter will only deal with the integration of tools that run (runtime execution) on a server/computer that can be reached over a network connection (internet).

As Deliverable D3.3 of STREAMER already stated, the objective of this project is to bring together the BIM and the simulation worlds. The principle that engineers would continue to use the software they use for their performance and quality rather than change for software they have not yet mastered still stands. So, the proposed approach aims to foster a "smooth" transition to integrate the practices of engineers into a BIM workflow rather than imposing a particular tool.

There are four options to integrate online tools:

- Centralisation into one tool
- Centralized data with distributed tools (one to one connections between data and tools)
- Distributed data connected via a central tool (APIs and BUS systems)
- Distributed data and distributed tools (Standardised API)

3.1 Centralisation in one tool

Using the same analogy as with data integration, the most basic way to integrate tools is to integrate them into one (new) tool. This creates a black box with as many features as possible.
This single tool or group of tools from the same family is often seen as the most efficient way to programmers. The conditions are very controllable and the result of the tool is very predictable. Internally, all the different data structures are mapped to the one (usually proprietary) internal data structure of the tool.
Experience (and research) has shown us that the AEC industry is so fragmented and specialized, that a tool (or even group of tools) can never serve the bulk of the industry.
This approach is not acceptable in the industry and therefore also not feasible for STREAMER.

3.2 Centralized data with distributed tools

There are many misconceptions about BIM. Maybe the biggest misconception is that BIM is centralizing all the data of a project into a single data repository. Because most BIM software tools work with a central database that is being used for all features, this concept is being copied on a project scale. The recent rise of online BIM collaboration platforms is feeding this concept. Many online BIM platforms are being developed at the moment. All of them have specific features and both positive and negative points.

A well-known initiative is the open source BIMserver.org project. This initiative started in 2008 when the industry was mostly unaware of the concept of online BIM. The stability of the BIMserver.org platform is proven to be enterprise ready. Many commercial applications build on the stable base of BIMserver. STREAMER partner TNO is heavily involved in the development and use of BIMserver.

Using BIMserver (or any other online database for that matter) as a single source of data for several online applications gives many advantages. The data is always in sync between partners and disciplines; users always work with the latest version of the data; etc.
However, BIMserver research has shown that collaboration processes using a central data repository are actually decreasing productivity of the project. Working efficiently with distributed data storages is more effective for the project than trying to centralize everything [Berlo2012]

Within the STREAMER project, the diversity of the data formats and standards also limits the flexibility of a solution based on a single data model.
Therefore, this solution is not seen as feasible within the STREAMER solution space.

3.3 Distributed data connected via a central tool

Different online BIM services from Autodesk, Bentley, Tekla, Graphisoft, Nemetschek, Onuma and others support different APIs. Connecting BIM applications directly, requires n*(n-1)/2 custom interfaces.

To limit the number of one-to-one connections the concept of an ‘Enterprise Service Bus’ (ESB) is introduced. An ESB is implementing a communication system between mutually interacting software applications in a service-oriented architecture (SOA). It implements a software architecture like this:

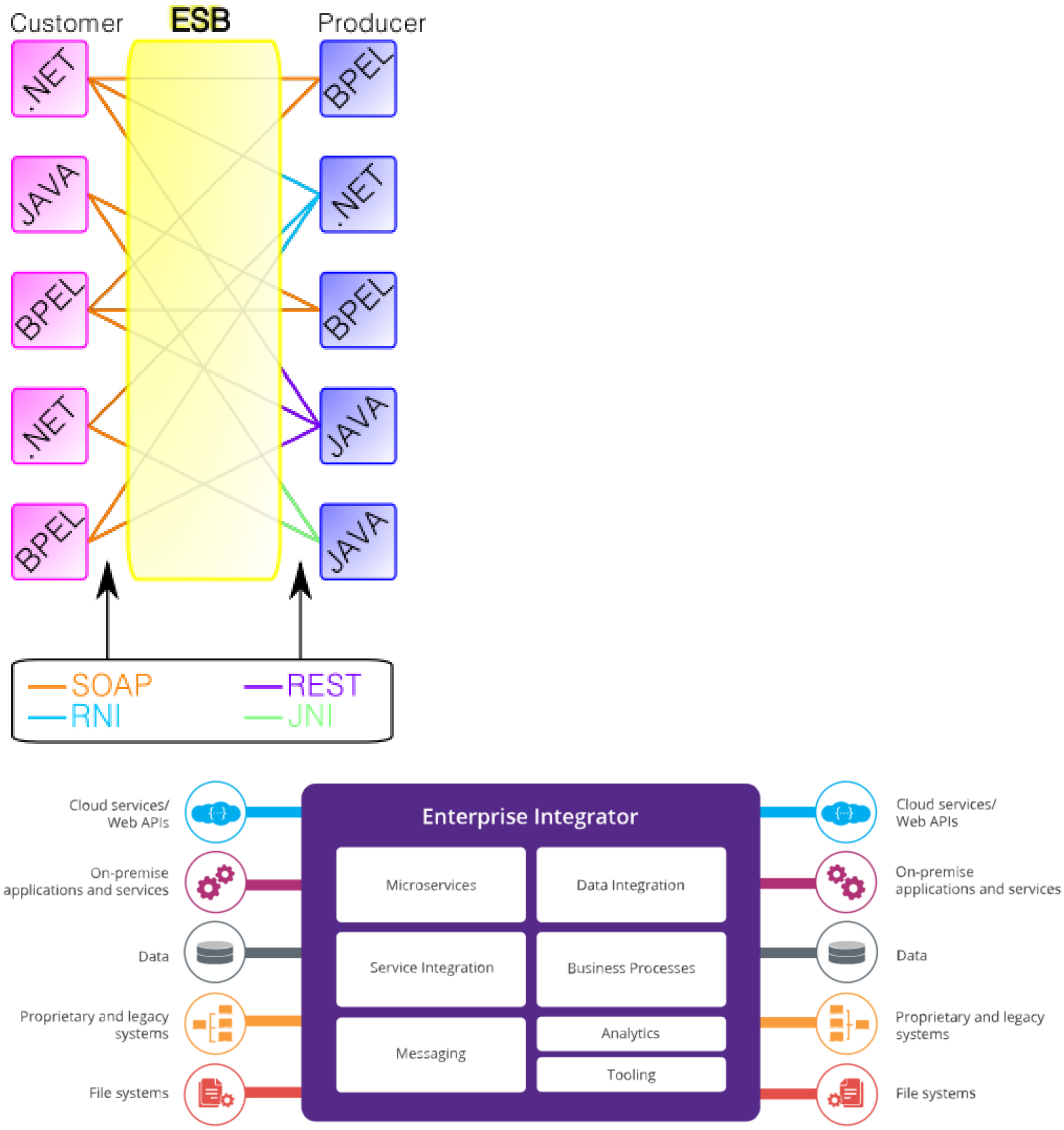


Figure 1: principle of an ESB (above) and the WSO2 Enterprise Integrator feature overview

As it implements a software architecture for distributed computing, it also implements a special variant of the more general client-server model. Whereas in general any application using ESB can behave as server or client in turns, ESB promotes agility and flexibility with regard to high protocol-level communication between applications.

An ESB limits the number of one to one connections since every application is connected to the ESB and the communication is streamlined by the ESB. Advanced ESB systems can also validate, aggregate, check and transform the data before forwarding or distributing it to connected tools.
STREAMER Deliverable 3.3 identified the need for intermediate tools to transfer data from a BIM format to the simulation tool by:

- checking / controlling data (including geometric data)
- assigning properties from external data sources BIM (material properties, use and occupation of the areas, equipment parameters)

An ESB could be used to facilitate this.

The use of an ESB to implement an improved data integration platform and open server for data and tools is very feasible and we are exploring the use in an experimental way.

3.4 Distributed data and distributed tools (standardised API)

Where ESB solutions are very useful to connect legacy systems in a given environment, the one-to-one connections to the ESB are still very labour intensive and limited to the API competences of the tools.

Implementing effective BIM and GIS data in the cloud requires automated (machine-to-machine) interaction between different online BIM services. A standardized web services API for online BIM services would enable machine-to-machine exchange of BIM data and enable innovation in the industry. BIMSie is intended for this, being the open web services API for online BIM services [BIMSie2017].
The recent developments around BIM Bots use a simplified version of the original BIMSie API to connect online tools.
Chapter 2.4.3 of STREAMER Deliverable 6.5 recommends the evaluation of the BIMSie API and that is what has been done. The latest developments around BIM Bots seem to be very promising since they create the possibility to automate connections between online tools.

The BIM Bots (updated BIMSie concept) therefore seems very feasible for a STREAMER solution for improved data and tools integration. However, STREAMER has to consider using different data models in a combined manner in order to realize the intended workflow. In Chapter 2.4.2 of D6.5 the Web Feature Service (WFS) is indicated as important for STREAMER to access the data of the hospital district itself and the surrounding district.
The BIM Bots concept therefore has to be evaluated against the use of:

A) BIM data formats
B) GIS formats
C) the combination of BIM Bots API and WFS API.

4. Proposed solution

4.1 STREAMER context

Within STREAMER the PLM server plays a central role in the connection between several systems. PLM (Product Lifecycle Management) consists in centralizing all information regarding an industrial product. The PLM is intended to be the central server between all the different tools and data. All the documentation, models, computation results will be stored on the server, will be accessible at any time by any actor and will be visible through any solution that will implement the PLM API.

The principle is the following: any PLM client sends a request to this central BUS. Whatever the language or the type of demand, the Information Logistics can answer by soliciting the corresponding data server in the right language. The localization of the data is known by the directory server and the translation between BIM and GIS data is made possible thanks to the semantic contained in the PLMs server configuration settings.

The technology used by the PLM is the same as the ESB technology described in chapter 3.3 of this document. The standardisation of the API described in chapter 3.4 is comparable with the standard API of the PLM.

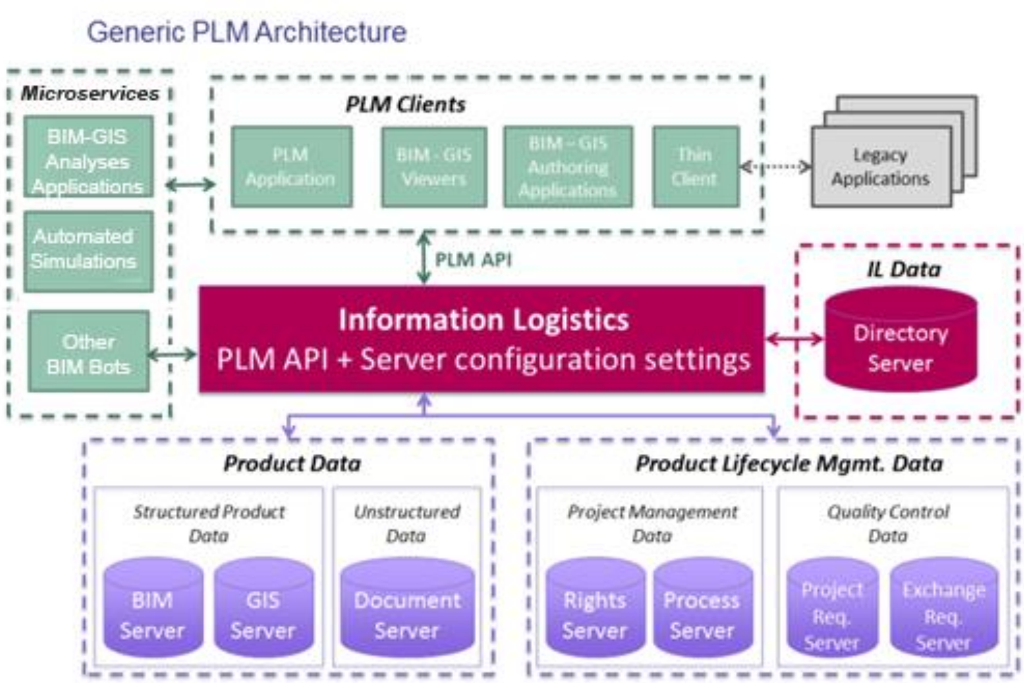


Figure 2: Generic PLM architecture in STREAMER

A PLM system coupling BIM, GIS and PLM concepts allows to formalize and to streamline the exchanges between several companies around the digital model BIM, the GIS spatial data and around all the data of the project, by protecting the management of the various actors' rights during the whole life of the project.

The PLM solution proposed in STREAMER D5.3 has been described regarding the inputs and outputs of the other work packages and its functionalities are described via an API detailed in D5.3. The objective of this API is to be as generic as possible in order to be able to connect to any PLM client in entrance and any type of data server in output. This intention is comparable with the concept described in chapter 3.4 of this document.

Nevertheless, there are some challenging issues:

- Being as generic as possible through the PLM API
- Being able to exchange data within various servers and to translate the data from a server to
- another one
- Determine a PLM tool that will enable to use the PLM API

There is a risk that no fully generic PLM API can be implemented. We think the solution will be to reach directly the good server that contains the data instead of going through the upper layout called generic directory server.

4.2 SOA approach to PLM

In practice, a PLM is a new system that tries to centralize all data and connect tools to a single server. An SOA approach to PLM must make PLM information available to any person or system that needs information and allow information flows to be orchestrated to support company-specific business processes. Ultimately, an SOA environment for PLM allows relationships between information in multiple different applications to be defined and managed to provide insight that has only been possible before when all data resided in a single system (IBM 2008).

Making the PLM available to other systems on the Information Logistics layer (the ESB part) creates the option to directly interact with other systems and data stores. This creates flexibility in the solutions and prevents a single point of failure in the whole network.

4.3 Overview solution approach

Generally, the STREAMER data can be split into three categories:

- The tools (simulations)
- The data (BIM, GIS and sensor data)
- The integration platform

Given the situation and analyses, we propose a combination of technologies to be used in STREAMER for an improved data and tools integration. The concept is sketched in the following image:

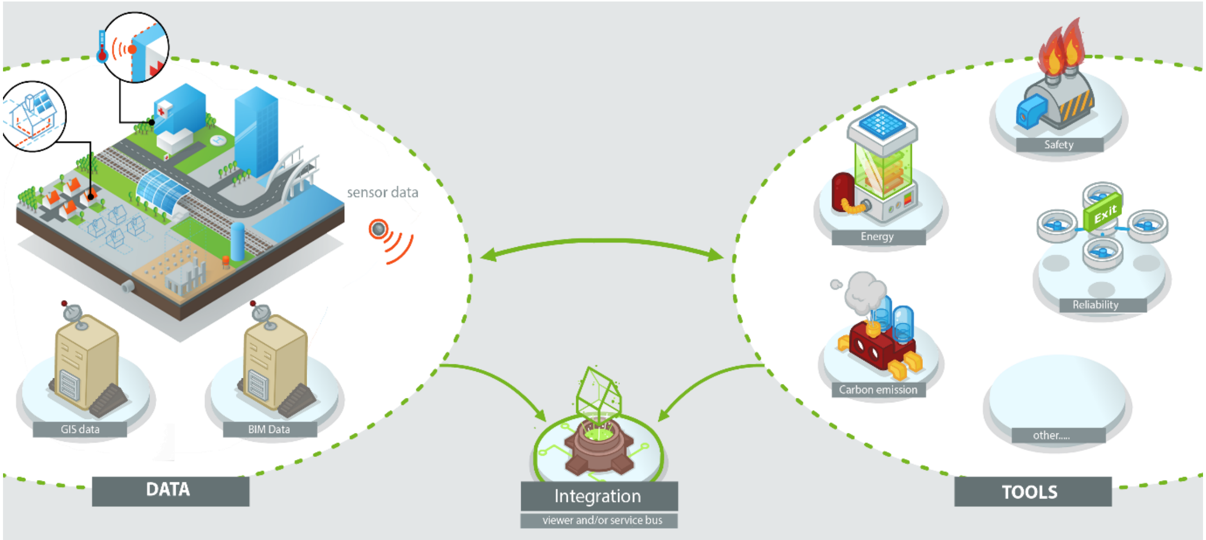


Figure 3: Integration principles of data and tools

The main difference between the conclusions of D5.3 and this report is that the use of PLM and a generic API to create a single point of integration comes with effects that are prohibitive to detailed integration of data.

Experiments suggest the use of PLM combined with a more open ESB approach when necessary to integrate the data and tools [Berlo2014]. The most promising solution to connect the several different entities is with an ‘event driven’ approach. This suggestion is seen as an additional functionality on top of the PLM approach.

4.4 Event Driven architecture

The basic principle of this concept is that the BIM data and the simulations are not lined up in a predefined workflow. The interaction between these tools is ‘event driven’.

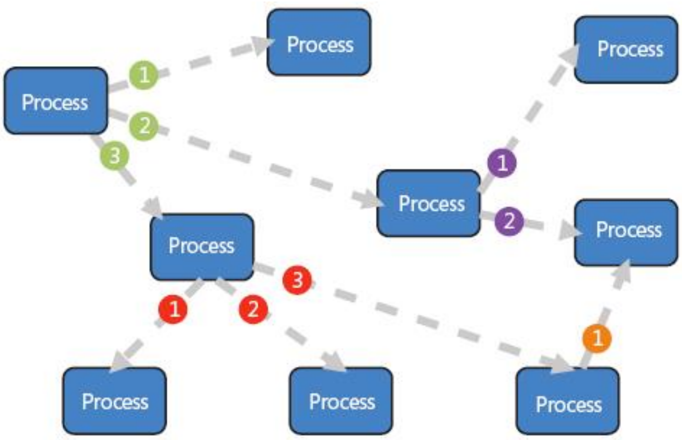


Figure 4: principle of event-driven architecture (source: Microsoft)

The principle of ‘event driven’ interaction is not new in IT architecture but has only been applied experimentally in the BIM and GIS applications. It facilitates the use of small ‘*microservices’* that seamlessly connect and integrate. These small services "do one thing and do it well" [https://en.wikipedia.org/wiki/Unix_philosophy].

The Wikipedia article on Microservices [https://en.wikipedia.org/wiki/Microservices] describes them as follows:

- The services are small - fine-grained to perform a single function.
- The organization culture should embrace automation of tasks.
- The culture and design principles should embrace failure and faults, similar to anti-fragile systems.
- Each service is elastic, resilient, composable, minimal, and complete.

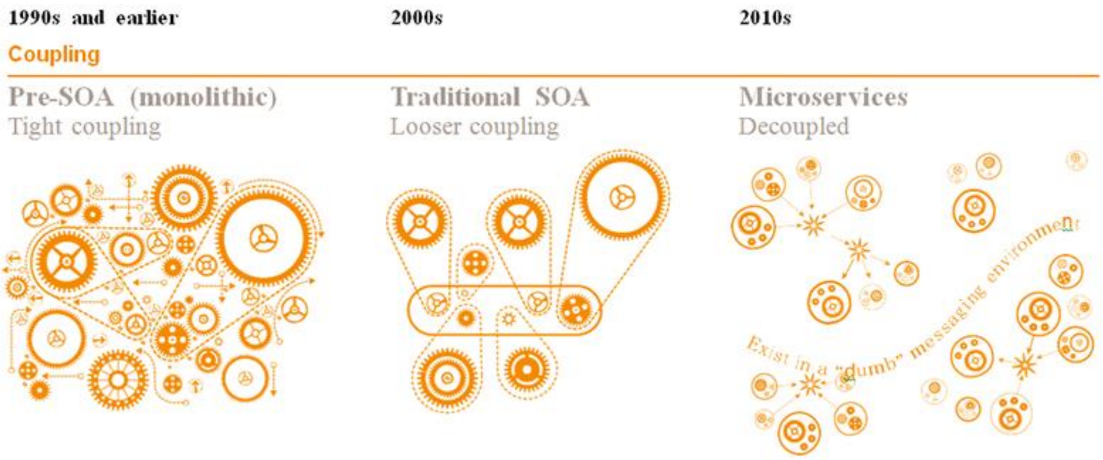


Figure 5: Linking 'microservices' in an event driven ecosystem (source: PWC)

The concept of stand-alone services is proposed on multiple events and deliverables in STREAMER. Every simulation tool should be developed as an online service that is minimal and complete.

This brings a challenge for the individual simulation tools, but is the most durable architecture for long-term innovations and business models.

5. Implementing

To demonstrate the SOA approach using microservices as proposed in the previous chapters, one new service was created in STREAMER context. The TECTool, which was developed by TNO, is converted into an energy calculation service. This tool uses multiple data sources; IFC, climate data, thermal properties of EeB solutions and properties of MEP systems. Adding a new BIM model to the BIM data service will initiate the energy performance calculation, the so called trigger.

The current implementation has only one service (BIM bot) and uses files for the remaining data sources (climate, thermal properties and MEP system properties), see figure 6. The files are currently hard-coded in the energy calculation service instead of configurable, which is fine as long as the service is a prototype.

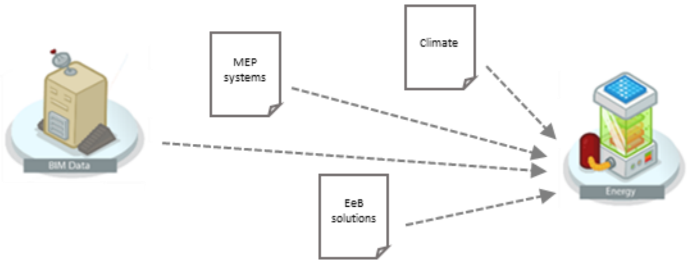


Figure 6: Energy calculation service using files for MEP systems, EeB solutions and Climate data.

The thermal properties and properties of MEP systems are not needed if this information was modelled in the IFC file. However, the way this information is modelled in IFC does not fit to the early design stage of STREAMER and how the energy calculation service needs it. EeB solutions are modelled in IFC via layers of materials where the energy calculation service only needs values for the combination of layers. MEP system parameters are in IFC not connected to spatial elements, but to distribution elements, which are contained in spatial units. Finding all elements in a system requires the complete network of MEP systems to be modelled, which is not expected in the early design stage. Therefor current models generated in STREAMER lack detailed thermal properties for building elements and HVAC system. By referring for building elements to an EeB solution and spatial elements to MEP systems, the relevant properties were found and energy calculation becomes feasible.

If the energy calculation service was truly a microservice the files would be replaced by other services (as in figure 7) and performed when needed to complete the BIM data necessary for the energy calculation service. The climate microservice should use geolocations of the BIM model to get the climate data and either insert this into the BIM model or store it as a separate file which is linked to the BIM model (extended data). For climate data, commercial services do exist [MeteoNorm], which is close to what is needed, but lacks the connect ability to the BIM bot framework as is. Creating it for the prototype would take a few days, but does require a valid license, which TNO does not have.

The MEP system and EeB solutions should ideally be part of the original IFC file and if not present could be added by other microservices. Such services will apply different rules to determine the correct MEP system(s) or EeB solution(s) for the model or will generate multiple versions of a model with different MEP systems and/or EeB solutions. In both cases the relevant properties will be added to the IFC using the semantics that is intended by the IFC schema. Having a BIM model completed with climate data, MEP systems and EeB solutions, will initiate the energy calculation service.

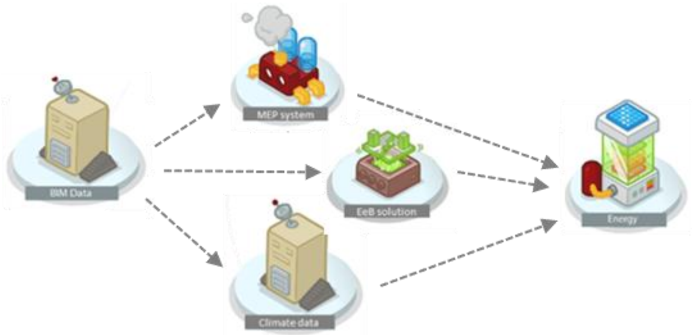


Figure 7: Energy calculation eco-system of microservices, the initial trigger is a new BIM model causing the services MEP systems, EeB solutions and Climate data to start. Only when a BIM model is completed with climate data, MEP systems and EeB solutions, the energy calculation service is started.

In the energy calculation chain of microservices, the MEP systems and EeB solution do not depend on one and another, but for energy calculation both have to be applied. Via checking mechanisms (mvdXML based) on the IFC file, this can be checked to make sure only models that are extended with climate data, MEP and EeB data will cause the energy calculation service to start.

The energy calculation service will calculate the amount of annual heating and cooling needs by a building, zone and space for a typical year based on EN/ISO 52016-1. Using the MEP system parameters (efficiencies) the needs are converted into consumed energy. Also the minimal and maximum power that is needed for heating or cooling are calculated. All these results are added to the BIM model and can be used in other BIM bots (or external tools) for analysis.

For energy calculations a number of behaviour parameters (number of occupants, ventilation amount, and temperature set-point on a daily schedule) are needed per space. In Streamer these are set via labels assigned to each space. The label represents a fixed set of values for these behaviour parameters and are more or less already part of the BIM model.

All though the energy calculation service as is implemented now has no need of other services, it can be connected to existing services. The connection to Zapier or WSO2 will help creating different chains of services, for instance more detailed and usable reporting of the energy calculation.

6. Recommendations

6.1 Background

Within STREAMER Deliverable 3.3 a list of energy performance simulation tools is gathered. Deliverable 6.5 had a strong focus on mapping data structures and standards. Deliverable 5.3 describes the PLM architecture.

Deliverable 5.3 also identified challenges of being as generic as possible through the PLM API, and being able to exchange data within various servers and to translate the data from a server to another one. To lower the risk of these challenges we propose to intensify the use of the 'Information Logistics' layer (the ESB) of the PLM.

Connecting tools directly to this layer of the PLM architecture improves the ability to integrate the data in a more semantic way.

6.2 Results and observations

At the moment we are using the WSO2 Enterprise Integrator [http://wso2.com/integration] and/or the Zapier Developers platform to connect with BIMserver.org for IFC data (with BIM Bots compliant API) to Deegree WFS server (with WFS API). The source code of this implementation is available on WSO2.com, Zapier.com, deegree.org and Github.com/opensourceBIM.

To increase the speed of prototyping, the developer platform of Zapier was used. Zapier creates the possibility to connect the STREAMER ecosystem with hundreds of tools like Google Spreadsheet, Gmail, Slack, dropbox, etc. This can also be used for rapid development, realizing that this creates dependence to this single online system.

6.3 Recommendations and future work

Additionally API managers and other tools could be used.

The work on the WSO2 Integration tool is progressing very fast. The concept of 'Bots' is also gaining traction in the industry. Connecting single online tools to each other using the microservice concept is the biggest research area in the field at the moment.

For STREAMER this approach is proven very feasible and based on the prototypes we recommend to continue with the approach.

Existing BIM tools could be converted to a single service. However multiple smaller services (if possible) are preferable. This way data created by a service can serve more other services.

7. References

[Bektas2016] Bektas, E., Brouwer J.: D4.5 IPD contract recommendations and best practices, STREAMER Deliverable of work package 4, August 2016

[Berlo2012] Van Berlo, L. A. H. M., et al. "Collaborative engineering with IFC: new insights and technology." 9th European Conference on Product and Process Modelling, Iceland. 2012.

[Berlo2014] van Berlo, Leon, and Thomas Krijnen. "Using the BIM collaboration format in a server based workflow." Procedia Environmental Sciences 22 (2014): 325-332

[BimBots2017] The bimbot infographic website describing what this is and how this works, http://bimbots.org, last access August 2017

[BIMSie2017] The official webpage of BIMSie at the National Institute of Building Sciences, https://www.nibs.org/?page=bsa_bimsie, last access July 2017

[Haefele2015] Häfele, K.-H., Benner, J.: D6.5 Advance Mapping Structures and Standards, STREAMER Deliverable of work package 6, February 2015

[Helm2015] Van den Helm, P., Häfele, K.-H., Benner, J.: D6.3 Energy modelling interface IFC – CityGML, STREAMER Deliverable of work package 6, August 2015

[MeteoNorm2017] Company offering commercial services to get TMY2 climate data relevant for energy performance calculations, http://www.meteonorm.com, last access April 2017

[Microservice2017] Microservice explained at Wikipedia, https://en.wikipedia.org/wiki/Microservices, last access August 2017

[Vial2015] Vial A.; D5.3 PLM architecture for EeB witihin the context of healthcare districts, STREAMER Deliverable of work package 5, August 2015

[WebService2017] Web service, Wikipedia, https://en.wikipedia.org/wiki/Web_service, last update 31. January 2017, last access February 2017

[WSO2_2017] WSO2 home page, http://wso2.com/integration, last access July 2017

[ZAPIER2017] Zapier Developers platform, https://zapier.com, last access July 2017

http://www.meteonorm.com