HBIM-GIS INTEGRATION WITH AN IFC-TO-SHAPEFILE APPROACH: THE PALAZZO TROTTI VIMERCATE PILOT CASE STUDY

Marzia Gabriele[1,*], Mattia Previtali[1]

[1] Department of Architecture, Built environment and Construction engineering (ABC), Politecnico di Milano, Via Ponzio 31, 20133 Milan, Italy

KEY WORDS: HBIM-GIS, data integration, ArcGIS, Shapefile, IFC, LOI, LOD, cultural heritage, preservation.

ABSTRACT:

The proprietary software investments in the data integration field are incrementing, and the progresses are visible in the possibility to directly open in a GIS environment a 3D software data format. Still, this is limited to the integration between the proprietary data formats and standards, ArcGIS environment shapefile multipatch and Revit 3D model, by using a proprietary software (ArcGIS). This study takes advantage of the lesson-learnt results in the proprietary data integration field, wanting to replicate a similar result using the IFC open standard, which is not directly openable by a GIS interface and needs to overcome a conversion that in most of the cases leads to semantic and geometric losses. So, an IFC-to-shapefile data conversion was performed, stressing (i) the way information is stored in the attribute table to query the geometries and perform geoprocessing, by (ii) implementing workarounds to keep the Revit instances' shared parameters in the IFC file, (iii) meanwhile having a high Level of Detail of the HBIM. The research performed the IFC-to-shapefile data conversion through FME (Feature Manipulation Engine), benefitting of the flexibility of the shapefile format and of the IFC' possibility to keep a high LOD in the export phase. Both allowed to properly query and manage the elements of an HBIM in a GIS (ArcGIS environment), and, using relational attributes table, retrieve the information contained in each Revit instance' property panel, as the shared parameters that implement the BIM Level of Information (LOI).

1. INTRODUCTION

In the vast panorama of data integration there are most recurring challenges deriving both from the use of open standards and of proprietary formats of exchange, and their consequential progressive development in the field. Speaking about the integration of two major framework, Geographic Information System (GIS) and Building Information Modeling (BIM), were initially conceived to provide digital representations of architectural or environmental entities, with different focuses. BIM is in the domain of Architecture, Engineering, and Construction/Facility Management (AEC/FM), giving attention to construction projects and their internal design, providing detailed micro-level 3D models that could be used throughout the lifecycle of a building (Zhu et al., 2018, Volk et al., 2014; Azhar, 2011). GIS analyses and visualizes location-related problems in geospatial science, mainly used to generate macro-level information, and deriving knowledge through various spatial analysis tools and modelling approaches (Longley et al., 2005; Ma, Ren, 2017; Doumbouya et al., 2017; Li et al., 2017; Zhu et al., 2018; Zu et al., 2019). The integration of these two technologies, implies a benefit in the future developments of the built environment both project and management, as it is not comprehensive to analyze building attributes and characteristics without taking into consideration the effects, pressures and impacts deriving from their surrounding environment (Wang et al., 2019, Zhu et al., 2018; Alsaggaf, Jrade, 2015; Ma, Ren, 2017; Wang et al., 2019).

The present research collocates in the field of HBIM (Historical BIM) and GIS integration, with the aim to investigate a methodology bringing together the use of open-source scenarios, the Industry Foundation Class (IFC), and the best practices in the field of proprietary software panorama (referring to the ESRI Multipatch Shapefile). The choice to use a *de facto* standard, as the shapefile, is because Open standard format for the 3D GIS environment CityGML, has proven to have multiple bottlenecks in the data conversion task, starting from the overall need to convert the solid 3D model (IFC) into a surface 3D model (CityGML) and the consequent need to overcome to the semantic transfer through a class mapping. ESRI desktop GIS applications (ArcGis Pro) is at the moment (2021) the one most investing in research for achieving the capability of ArcGis to be a complete data integration platform, fully delivering the (direct or indirect) exchange of readable data formats in the ArcGis software (ESRI, 2020); (i) direct: through the implementation in the ArcGis Pro software (from the 2.7 version) of the possibility to open a Revit 3Dmodel, operating a folder connection and dragging it into the GIS environment, this workflow preserves the Revit 3D model geometry and semantic data organization, gives the immediate possibility to query the single geometries and to get information and characteristics of the building, allowing basic operations mainly related to the building itself and not to its single layers, as the information are organized in sub-categories of discipline layers; (ii) indirect: through the use of the ArcGis Pro DIA (Data Interoperability Extension), an integrated spatial extract, transform, and load (ETL) toolset that runs in the geoprocessing framework using Safe Software's Feature Manipulation Engine (FME) technology that allows to perform a data format conversion through a no-code integration approach in ArcGIS.

The first part of the article makes an overall summary of what are today's main bottlenecks, issues and problems when dealing with HBIM-GIS integration and the potentials that could derive from it. Then, literature review focuses on the most recurrent open and proprietary standards: IFC, CityGML and ESRI Shapefile. The second part of the work is dedicated to a pilot application of HBIM-GIS integration, with the case study of Palazzo Trotti in Vimercate (Italy); a portion of the palace was (i) modeled using the software Revit (Autodesk), then (ii)

https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-167-2021 |

exported into the IFC format through the IFC Exporter plugin for Revit, mapping the Revit families to IFC Classes. The data conversion and attribute retrieving workflow has entirely been developed in the FME (Feature Manipulation Engine) environment. The advantages of using the here presented IFC-to-shapefile research approach to pursue an HBIM-GIS integration, are in the possibility to (i) define the LOD (Level Of Detail) of the IFC model directly in the Revit user interface, overcoming the LoD (Level of Development) of the CityGML model; (ii) directly achieve a geometric conversion, from solid (IFC) to solid multipatch geometries (shp), with no need to deal with the solid-to-surface issue and semantic mapping of the CityGML model; (iii) perform a join of the single instances' attributes of the Revit model by the IfcGuID belonging to each correspondent mapped class of the IFC through shapefiles' relational database.

2. HBIM-GIS INTEGRATION AND STANDARD FORMATS

2.1 Trends and approaches in the current HBIM-GIS integration panorama

Even if the HBIM-GIS integration is relatively recent (Ramirez Eudave, Ferreira, 2020), we can benefit from the experiences reported in the literature, that suggest auspicious methodologies and approaches in the field of management and safeguard of the built cultural heritage. Matrone et al. (2019) develop a methodology for importing the HBIM model into a GIS environment, both commercial and open source (ArcGIS Pro and QGIS), using FME for converting the IFC data format into the CityGML format, to obtain a unique and unified model and vocabulary for the 3D GIS project, structured by LoD standard. Campanaro et al. (2016) discuss the set-up of a novel three-dimensional system for the management of heritage structures by exploiting the combination of 3D visualization and geographic information systems (GIS) analysis for creating a fully queryable digital twin made with GIS features (Ramirez Eudave, Ferreira, 2020). De Ruvo (2019) explores the interoperability of HBIM-GIS frameworks in the cities of Quinson, Tolentino and Venice; the work, developed in the framework of the ResCult project, move from disaster management to the development of a disaster risk reduction strategy; the approach follows the (Matrone et al., 2019) method as the BIM-GIS integration occurs within the multiscale representation standard provided by CityGML in an open-source environment (QGIS) through FME conversion. Lastly, the work by Vacca et al. (2018) focused on "La Gran Torre di Oristano", aimed at the development of a workflow able to integrate the 3D model of the "Gran Torre" into ArcGIS and ArcScene; the work was finalized at the contribution of the HBIM and GIS methodologies in the structuring and management of a wide range of digital data and information useful for its management. The fact that there is no unique procedure, makes the real-world integration of the two systems difficult, as it still presents several obstacles such as the different ways of representing geometries, the use of different datums and coordinate systems, and the different semantics used (Vacca, 2018).

2.2 The open and proprietary standards in the panorama of BIM-HBIM-GIS integration

BIM and GIS use diverse data schemas for data storage, management, and exchange, due to their roots in two distinct areas (Amirebrahimi et al., 2016; Zhu et al., 2018; Liu et al., 2017). Neither BIM nor GIS can effectively and directly read data from the other. As a part of BIM/GIS data integration, data conversion is usually involved, this is necessary to bridge from the abovementioned ones (GIS and BIM) to a third system (Ma & Ren, 2017). Among those conversion patterns, the mainstream of data conversion is from BIM to GIS, which means to convert BIM, including semantic information and geometric information (Iiskdag et al., 2007; Borrrmann et al., 2015), to the GIS world using open and proprietary standards. For example, (i) in Irizarry et al. (2013) the import of an IFC file to GIS environment was performed in the ArcGIS desktop by ESRI Multipatch Shapefile (ESRI, 2008) whereas a central database (e.g. MS Access) is used for transferring attribute data between BIM and GIS; (ii) in Jusuf et al. (2017), the theoretical research used a FME-guided approach to restructure the data model (IFC) to transform it to CityGML.

2.2.1 BIM framework: the IFC

The Industry Foundation Classes (IFC) is the representative data format for BIM. IFC is open-source standard format (ISO 16739:2013) widespread in the Architecture Engineering Construction domain. It is object-oriented and fulfills the need for 3D data interchange between the vast panorama of parametric modeling software. As for that purpose, the IFC can store all kind of information related to the different phases of the project, for each part of the building. The LOI (Level of Information) related to the geometry can broadly go from e.g., the material cost during the project phase to the building doors condition assessment during the maintenance works. That is to say that the flow of information carried in multiple situations related to the AEC field can be stored and accommodated in the IFC, making it become the most neutral open format to exchange information.

2.2.2 GIS framework: CityGML and shapefile

The XML-based format CityGML is the main open data model used for storing and exchanging information related to virtual 3D city models, it is a Geography Markup Language (GML), an international standard format defined by the Open Geospatial Consortium (OGC) and Geography Markup Language version 3.1.1 (ISO TC211). The aim of the development of CityGML is to reach a common definition of the basic entities, attributes, and relations of a 3D city model (OCG, 2020). In CityGML different geometries can be associated within the same object, to have a multiple representation, which can be based on different eras, different construction hypotheses or different levels of development (LoD). Regarding IFC-to-CityGML conversion, IFC mainly uses solid modelling methods, such as CSG and swept solid (Sacks et al., 2018) whereas CityGML mainly use surface models (OCG, 2020), which does not support volumetric modelling of walls, roofs, or slabs (Geiger et al., 2015). For example, in IFC, a piece of wall is represented by a volume, but in CityGML, the same wall is probably represented by two surfaces, e.g., one for representing interior wall and the other for exterior wall (Isikdag et al., 2013). This founds its explanation in the fact that CityGML it is used to record observable objects it is much 'closer' to surveying and photogrammetric registration methods than BIM (Nagel et al., 2009).

The option deriving from a "*de facto*" applicative use, is the ESRI shapefile (Isikdag et al., 2013; Zhu et al., 2019). Shapefile is native to GIS environment and is used to store geometries and attribute information (Brundu et al., 2014), today is one of the most common formats for geospatial information exchange and surely has a huge impact in the GIS environment panorama.

Shapefile stores in the datasets both the geometric and semantic information related to the spatial features. Each shape has a one-to-one relationship within its correspondent attribute. Geometrically, the shapefile can host 3D geometries by the means of the MultiPatch and has the advantage of allowing rapid modifications of the 3D elements directly from GIS user interface, meanwhile demanding less effort in terms of disk space and writing. Shapefile allows to overcome the issues related to the CityGML format and its conversion workarounds, e.g.: (i) the need to follow predefined a LoD, since a shapefile allows a simpler definition of user-defined data model according to the specific application and user-needs with specific objects and relationships among them; (ii) the required transformation from a IFC solid to a CityGML surface model and its consequential semantic mapping, as shapefile is itself both a solid and surface model, capable of directly accommodating 3D IFC geometries (Deng & Cheng, 2019).

2.3 The IFC-to-shapefile approach

The IFC-to-shapefile approach uses the shapefile in the specificity of multipatch geometry feature; this is a methodology to depict 3D features in commercial GIS that has progressively got the attention of the GIS community. In the panorama of data integration of proprietary software and standards, the ESRI software ArcGIS Pro, owing a huge plus to the use of multipatch shapefile, is currently the one more investing to achieve a good data integration between BIM and GIS. Today (2021), it allows directly from the ArcGIS 2.8 user interface both to (i) open a Revit model in ArGIS, creating a folder connection to the .rvt file itself and to (ii) perform a data conversion from open standards (e.g., IFC) to the multipatch shapefile through the DIA (Data Interoperability Extension, ref. to par. 4.1). It is of outmost importance to understand the level of integration reached by ESRI within the Revit-to-shapefile approach, which was used as a *best practice* reference for this study. ArcGIS software allows to open the .rvt BIM model without any losses on both the physical elements and semantic aspects of the building; as in fact when the Revit 3D was inserted into the ArcGIS environment, it completely preserved its geometry and semantic data, allowing to easily access the geospatial information, perform queries and achieve a proper management of the HBIM itself; these are the aspects that the research tried to replicate as a best practice applied to the IFC-to-shapefile approach, finding workarounds for the IFC framework, that could allow to best exploit the geometric and semantic benefits deriving from the conversion to shapefile. On one hand, from the geometric point of view, the data integration through multipatch shapefile depicts the 3D elements within a polyhedron technique, which is defined by triangle-based OpenGL 3D primitives, and where the features are stored in a geoDB as "true" 3D geometries. A multipatch can represent a wide variety of elements, going from elaborated to basic geometries, linked to X, Y, Z real-word coordinates. Technically, the mutipatch shapefile can accommodate the 3D IFC geometry with no required solid-to-surface conversion. On the other hand GIS hard core is to perform analysis and queries to give solutions to difficult interrogatives. To do that, GIS uses the geoprocessing tool, relating information (attributes) to the geographically positioned geometric features, by the means of attribute tables. Thus, it important to stress the fact that, born for a GIS-oriented environment, the multipatch composition (.shp, .shx, .dbf) allows to perform those analysis and fit within the complexity of the GIS software geo-analytic sovrastructure. The relational database technique behind shapefile enables it to store, extend and query IFC semantic information; if the LOI (Level of Information) semantics needs to be extended, additional attributes can be extracted from IFC and linked with the model through the Global ID, as in the case of retrieving the Revit shared parameters.

3. THE CASE STUDY HBIM

3.1 The case study: Palazzo Trotti Vimercate (Italy)

Trotti palace was built in the second half of the 17 Century by Count Giovanni Battista Secco Borella and his son Franceso, who were the main landowners of Vimercate. The new palace of the family, later indicated as Palazzo Trotti (Figure 1), was started by Giovanni Secco Borella as demonstration of the influence that the family reached in its feud. In 1701, after his death, the building was indicated as complete. Further works, like internal decorations by fresco technique, were then carried out by the son, Franceso Secco Borella. At that time, the Trotti family started to gain an important role in politics, also in military and religious field. According to the historical chronicles, the services of Antonio Trotti increased the importance of Trotti family and earned the respect of Giovanni Bentivoglio, Lord of Bologna, who inscribed Antonio Trotti to his family, giving origin to the branch Trotti-Bentivoglio. In 1716, Gian Battista, belonging to a second branch of the family, married with Giulia Secco Borella, daughter of Giovanni Secco Borella, feudatory and count of Vimercate. Therefore, the feud passed to Trotti Family, until 1796, when Napoleon moved to Italy and suppressed the feudal privileges. The building takes its name from the Trotti family, who were its last owners before it passed into public ownership (Penati, 1957; Bossaglia et al., 1990). Since 1862 the building has been Vimercate's Town Hall.

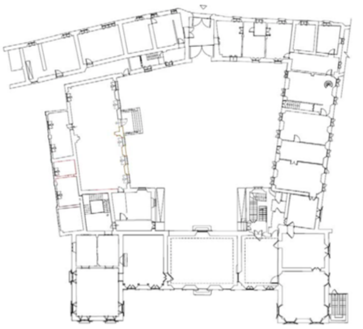


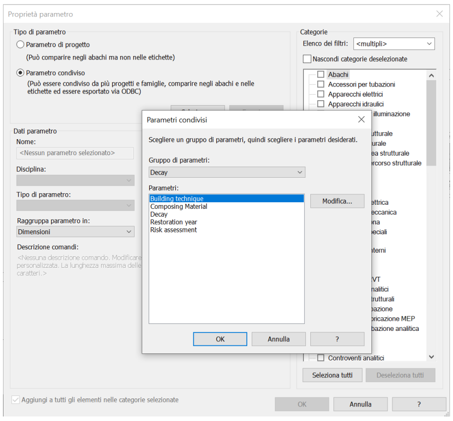
Figure 1. Plan of the Trotti Palace Groundfloor.

3.2 The Information model of Palazzo Trotti Vimercate

The case study of Palazzo Trotti in Vimercate developed the HBIM parametric modelling on the frontal-west wing portion of the palace. The decision was to organize all the elements of the Historical Building into the defined Revit categories to structure an HBIM that could intrinsically follow the logic of the software itself; moreover, having a precise organization of the elements of the model reveals to be outermost fundamental when mapping the IFC elements, because it allows an easier retrieving of the geometric information when in the FME environment. In addition to this, this first HBIM-GIS integration was conceived as a pilot test of the whole IFC-to-shapefile conversion. Even if guided by pilot-testing purposes, the modelling phase exacerbated the complexities of the Revit software itself, as this choice implicated to understand how to adapt the modelling strategies the context of historical building representation. So, having as a reference the 2D survey, the parametric model was developed with the BIM Autodesk Revit software, with some chances of openness to flexibility. The architectonical elements as walls, floors and the roof were defined using "system families", the main predefined typologies that can be found in the Revit software. Then, to model elements specifically belonging to the HBIM, as e.g., Windows and Doors, there was the possibility to create new external .rfa files of Windows and Doors families and import them into the project as "loadable families". Also, those constructive or structural elements unique of the complex historical building (e.g., the external stairs of the entrance, the historical walls having irregular shapes and variable thickness) that were not possible to model both within the system nor the loadable families, were modeled though the use of the "model-in-place" tool; with "in-place" option, local instances, belonging to component families, are created into the current project, but with the disadvantage of not having the possibility to be transferred nor used in other projects, also, if those elements are repeated many times into the model, the file becomes heavy to manage. Nevertheless, this is a good compromise that allows to reproduce those unique elements and components of the Historical building, just by using commands from the Revit interface.

3.3 The shared parameters of the HBIM in Revit

Dealing with transferring HBIM attributes within the shapefile framework and the consequential need to create and export attribute tables from the 3D software. which means to create schedules in the Revit software, shared parameters are essential (Figure 2). Following the workflow, (i) creating a shared parameter, (ii) adding it to the desired family categories, (iii) create a schedule with these categories (multi-category schedule in Revit), allowed to achieve the attribute storage and export through the .txt format.



Proprietà parametro

Tipo di parametro

○ Parametro di progetto
(Può comparire negli abachi ma non nelle etichette)

● Parametro condiviso
(Può essere condiviso da più progetti e famiglie, comparire negli abachi e nelle etichette ed essere esportato via ODBC)

Dati parametro

Nome: <Nessun parametro selezionato>

Disciplina:

Tipo di parametro:

Raggruppa parametro in: Dimensioni

Descrizione comandi: <Nessuna descrizione comando. Modificare personalizzata. La lunghezza massima delle caratteri.>

Categorie

Elenco dei filtri: <multipli>

□ Nascondi categorie deselezionate

□ Abachi
□ Accessori per tubazioni
□ Apparecchi elettrici
□ Apparecchi idraulici
□ Controventi analitici

Parametri condivisi

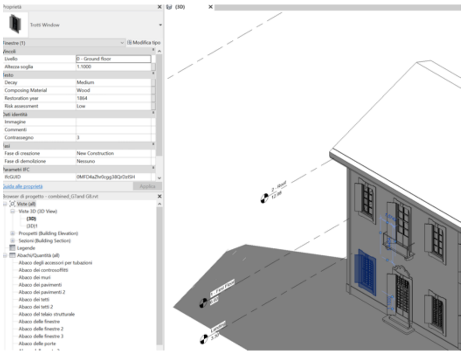
Scegliere un gruppo di parametri, quindi scegliere i parametri desiderati.

Gruppo di parametri: Decay

Parametri:
Building technique
Composing Material
Decay
Restoration year
Risk assessment

Figure 2. The creation of the shared parameters.

Shared parameters were chosen for carrying the HBIM information, as it mainly means to deal with something that mostly cannot be described with pre-existing attributes. Adding more complexity to the semantics of the HBIM, allows to achieve a good level of information (LOI) regarding the Cultural Heritage properties (HBIM). In this way it will be possible to archive and then retrieve information about specific properties of the heritage, here the study didn't pretend to define an ontology for the HBIM, but to effectively achieve its attribute storage and export through the use of Shared Parameters groups, e.g., (i) a group named "Historical Materials" and then add parameters as: "H-Material" (individuating the historical material itself); "Techniques" (if it is made with a particular technique, e.g., Capriata lombarda); "Year of construction" (...); (ii) a group named "Decay" and then add parameters as: Decay (ICOMOS); Composing Material; Restoration year; Risk assessment; Building technique (...). The benefits of using Shared Parameters instead of the Project Parameters for adding new information to HBIM, apart from the abovementioned issues, relies in the fact that when using the second ones, it shows up a problem of closeness of the information within external projects, so if e.g., we want to create a shared common knowledge for the HBIM, replicable in other HBIM scenarios having common features, it won't be possible in any way to retrieve single information coming from the project, and there will be the need to re-issue the attributes from zero: e.g., if having the information on the year in which a single component was restored included in the project as a "project parameter", there will always be a need for the creation of another "project parameter" for retrieving that information in another project. That is since a Project Parameter works as a container of information, that is defined and added to multiple categories of elements within a project. It has the peculiarity of being specific to the project to which it refers and in not being able to be shared with other projects or families. To overcome this problem, it is necessary to use the Shared parameters, as they can be used in multiple families and projects; the .txt file stores the information of the project in a file that is independent from them, allowing to retrieve the same information when needed for another HBIM project. From the building management point of view, when creating a new Shared Parameter, the "Instance" option must be preferred instead of "Typology" option, as in this way the project parameter is associated with the single element itself (Figure 3), belonging to a certain family. In this way it is possible to give to all the project elements belonging to the same family a specific attribute and then retrieve them with the use of multi- or single-category schedules.



Proprietà

Trotti Window

Finestre (1)

Vincoli	
Livello	0 - Ground floor
Altezza soglia	1.1000
Testo	
Decay	Medium
Composing Material	Wood
Restoration year	1864
Risk assessment	Low
Dati identità	
Immagine	
Commenti	
Contrassegno	3
Fasi	
Fase di creazione	New Construction
Fase di demolizione	Nessuno
Parametri IFC	
IfcGUID	0MFD4aZhr0cgg38QrDzfSH

Browser di progetto - combined_G7and G8.rvt

Viste (all)
Viste 3D (3D View)
(3D)
(3D)1
Prospetti (Building Elevation)
Sezioni (Building Section)
Legende
Abachi/Quantità (all)
Abaco degli accessori per tubazioni
Abaco dei controsoffitti
Abaco dei muri
Abaco dei pavimenti
Abaco dei pavimenti 2
Abaco dei tetti
Abaco dei tetti 2
Abaco del telaio strutturale
Abaco delle finestre
Abaco delle finestre 2
Abaco delle finestre 3
Abaco delle porte

Figure 3. The association of the shared parameters with the single instance of the Revit model.

4. MATERIALS AND METHODS

4.1 IFC-to-shapefile approach

There are not too many tools available for the transformation of IFC to shapefile. According to literature, Feature Manipulation Engine (FME) and Data Interoperability for ArcGIS (DIA) are the most frequently used commercial tools; another potential tool is Open CASCADE. Open CASCADE is an open-source software development kit (SDK) mainly used in areas such as computer-aided design (CAD), computer-aided manufacturing (CAM) and computer-aided engineering (CAE) for processing and visualizing 3D objects (SAS, 2019). In this framework, the investigated IFC-to-shapefile methodology aims to satisfy the requirements that most concern carrying an HBIM Condition Assessment (CA) in a GIS environment for Cultural Heritage management, addressing the practical issues that derive from importing an IFC model to a Proprietary GIS environment (ArcGIS). Stated that the digitization of Cultural Heritage is an essential prerequisite for the proper conservation, management and enhancement of structures (Massimiliano et al. 2021), the main requirement of the research was to define a HBIM-GIS rapid conversion methodology, mainly devoted to its applicative use, to help storing the CA information of the HBIM in a geographic context, useful for a rapid implementation and easy to fill out when carrying e.g. an on field survey of the historic Trotti palace, but that could be also useful for multiple actors that want to access this kind of information to carry out the management work, as electricians, carpenters etc. The requirements for the BIM-GIS conversion were to (i) rapidly integrate the HBIM in a GIS environment meanwhile avoiding geometric losses and cascade effects (e.g., problems deriving from 3D spatial intersections); (ii) overcome the semantic losses of the conversion by keeping all the attributes of the Revit file (shared parameters).

4.1.2 IFC-to-shapefile through DIA

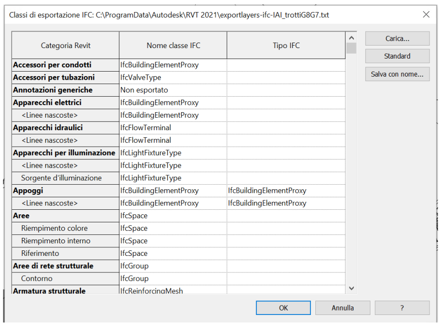
The study for the IFC-to-shapefile conversion was firstly conducted by testing the potentials of the integrated ArcGIS Data Interoperability extension for Desktop (DIA). The DIA is an integrated spatial ETL toolset for the ArcGIS software, that makes possible to perform multisource data integration directly from the ArcGIS Pro user interface. The DIA comes with a data engine, which is the Safe Software's Feature Manipulation Engine (FME) technology, and it supports the integration of a vast number of data-formats (i.e., IFC, CityGML). The DIA is more often used than the FME itself because its user interface is much easier to use and understand as it is based on a partially full automated conversion between the formats (Zhu et al., 2019). Still, performing the conversion using the DIA the problems were evident, as lack of a total control of the data conversion process from the interface leads to multiple errors, resulting in a qualitatively scarce IFC-to-Shapefile conversion. The following are the issues more related to the use of the DIA in literature: (i) a resultant execution error due to a wrong setup of the data exchange criteria; (ii) geometric information error in the middle of the process, resulting in a data damaged semantically and geometrically. In terms of Level Of Information (LOI), the semantic extracted from IFC data with this methodology, registers a loss in attribute organization and detail (also no shared parameters were retrieved from the Revit), defining an overall poor performance. IFC-to-shapefile conversion has never been regulated with standards, the literature dissertations in this field are mainly exposing procedures derived from the use of the shapefile in the pragmatic applicative field of research. This defines the need to pave the way for a common-agreed conversion process.

4.2 From HBIM to 3D GIS

Stating the problems that derived from using the DIA, there happens to be the need to pursue a methodology that overcomes both the semantic and geometric losses, and in the meantime performs a conversion from a HBIM 3D model to the ArcGIS Pro environment that could in a way resemble the results obtained in the field of ESRI proprietary software. So, the workflow was then oriented to perform a data conversion by using the FME (Feature Manipulation Engine) Software. The rest of this paper is about developing a framework within the use of FME for converting IFC file to ESRI multipatch shapefile, both considering LOD and LOI of the IFC, succeeding in achieving also a good semantic database migration from the BIM 3D software to the GIS. The workflow is based on the following steps: (i) create the HBIM in the Revit environment, using the families offered by the software to organize the elements (instances) of the model itself -where not possible the "mass" element was the one used-; (ii) export the HBIM through the IFC classes mapping (by using the IFC exporter plug-in for Revit 2021); (iii) perform the conversion for each of the IfcClasses to a .gdb (Geodatabase) through FME software transformers; (iv) retrieve the attributes by using the IFC GUID through FME *DatabaseJoiner* transformer; (v) reproject the geometries to fit in the current ArGIS workspace; (vi) open the results in the ArcGIS environment to test the performance of the workflow in terms of interrogability (e.g., spatial query); (vii) improve the management of the HBIM categories through PyCharm interface. This methodology successfully overcomes the DIA geometrical transformation issues, successfully achieving the geometric conversion without running geometric information execution errors or losses.

4.2.1. HBIM to IFC semantic mapping and export through the Revit IFC plugin

The first step of the workflow was to set a .txt file to properly perform the conversion between the Revit categories and the IFC classes. This was of outmost importance, as those categories are then being kept in the structure of the attribute table, to perform, when in the ArcGIS environment, semantic queries of the BIM based on the grouping of elements having the same characteristics.



Classi di esportazione IFC: C:\ProgramData\Autodesk\RVT 2021\exportlayers-ifc-IAI_trottiG8G7.txt

Categoria Revit	Nome classe IFC	Tipo IFC
Accessori per condotti	IfcBuildingElementProxy	
Accessori per tubazioni	IfcValveType	
Annotazioni generiche	Non esportato	
Apparecchi elettrici	IfcBuildingElementProxy	
<Linee nascoste>	IfcBuildingElementProxy	
Apparecchi idraulici	IfcFlowTerminal	
<Linee nascoste>	IfcFlowTerminal	
Apparecchi per illuminazione	IfcLightFixtureType	
<Linee nascoste>	IfcLightFixtureType	
Sorgente d'illuminazione	IfcLightFixtureType	
Appoggi	IfcBuildingElementProxy	IfcBuildingElementProxy
<Linee nascoste>	IfcBuildingElementProxy	IfcBuildingElementProxy
Aree	IfcSpace	
Riempimento colore	IfcSpace	
Riempimento interno	IfcSpace	
Riferimento	IfcSpace	
Aree di rete strutturale	IfcGroup	
Contorno	IfcGroup	
Armatura strutturale	IfcReinforcingMesh	

Figure 4. IFC mapping table from Revit interface.

To classify the categories of the Revit model into IFC Classes, dealing within the framework of the HBIM, the main problem concurred in understanding which element to map into what category to best fit the gap in-between the semantics of the Historical and not-Historical elements, for this purpose, the manual for the 2x3 IFC Standard was the one followed as a reference. Surely, there is no specificity nor in the Revit Categories nor in the IFC Classes that refer to the HBIM, so the aim of the work remains in the field of performing a good strict mapping of the Revit elements within the correspondent IFC Class (e.g., in the .txt file Structural framing – Revit - is mapped as IfcBeamType – IFC -), and when not possible, the option to use the IfcBuildingElementProxy was considered. This approach aims to standardize the Revit parameter names using the IFC mapping table (Figure 4); as a watchpoint of the methodology, the fact that the mapping of Revit System families is limited, and so, there is the need to overcome the minor issues and errors derived from a *forced* mapping when in the FME software. This first part of the workflow goes together within the second part of the IFC export, as in fact, to properly achieve the conversion, it is necessary to define some criteria when in the IFC export interface, here the ones used for this study: (i) define the LOD (Level Of Detail) of the model, which in a way gives the possibility to express the amount of details of the model; final exported model had a High LOD that could be visually checked e.g., in the IFC details of the doors and in the complex structure of the beam systems; it is important to remark that this is not a conversion *per se*, but is a conversion that is a first part of a workflow that aims at introducing a 3D model in a GIS within a .shp (shapefile) conversion workflow; as the shapefile doesn't rely on the LoD of the elements as the .CityGML file does (as it needs to overcome a LoD semantic mapping of the elements in order to perform the conversion), the choice of maintaining an High LOD in the .shp conversion relies partially on the sake of not losing those peculiar details of the elements modelled for the HBIM. Then, always through the user interface of Revit IFC export, (ii) the IFC GUID was stored as a parameter of the singular element belonging to each IFC class; this made possible to achieve, when in FME, the full join of the IFC single elements with the attributes extracted from the Revit instances, creating schedules for each of the. rvt families, the workflow here expressed wanted to keep the LOI of the original model, avoiding the semantic losses that occur when passing from the Revit to IFC.

4.2.2. From IFC to Shapefile through Feature Manipulation Engine (FME)

After the file was exported from Revit, it was opened in the FME Data Inspector software, to have a visual check and better understand if the conversion went well in terms of both keeping the geometries and if the attribute table of the IFC was storing the IfcGUID, this one was visible in the attribute table under the name of GlobalID. After this first check, the file was opened in FME Workbench, here the new workspace was set within the *Generate Workspace Dialog* interface, where the input data (feature Reader) was the IFC data format 2x3, and the output file (feature Writer) was the Geodatabase file (.gdb), with an output dimension of 3D+ Measures, a surface and solid storage of Multipatch shapefile, choosing to treat measures as elevations. The IFC geometric features chosen to be imported were: IfcBuildingStorey, IfcDoor, IfcRoof, IfcSlab, IfcWall, IfcWallStandardCase, IfcWindow, IfcMember, IfcBeam, IfcStair, IfcStairFlight, together with non-geometric feature, the IfcBuilding and the IfcSite. This workflow used the Geodatabase file format (.gdb), to retrieve the the multipatch .shp solid element 3D geometry. The extraction of the 3D geometry was exploited by the *GeometryFilter,* using the *Solid* geometry by the *GeometryExtractor* transformer. After the successful retrieving of the 3D geometries for each of the IfcClass, a visual check of the elements was performed though the FME Data Inspector; from the checking what emerged was that some of the elements were not precisely mapped in the IFC Semantic Mapping performed though the Revit software IFC export interface; as mentioned before, the most significant problem of the methodology relies on the fact that not all the elements of the Revit model can fit into a IFC classification and thus, mapping, so there are elements that after the exportation are in a formal wrong IfcClass; in the case study, those minor issues concerning the forced mapped elements were fixed by retrieving the elements by their Revit schedule attributes, then inserting them into the right converted Geodatabase file by a junction, e.g. (i) IfcSlab class containing the roof box end; (ii) IfcMember containing IfcBeam ISO 6707-1:1989 elements (Figure 5).

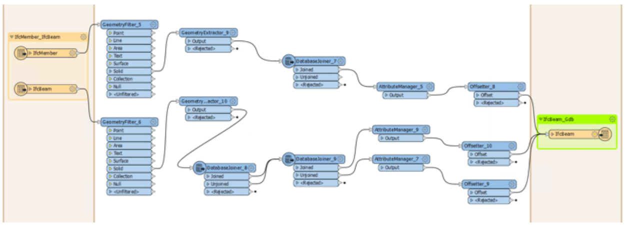
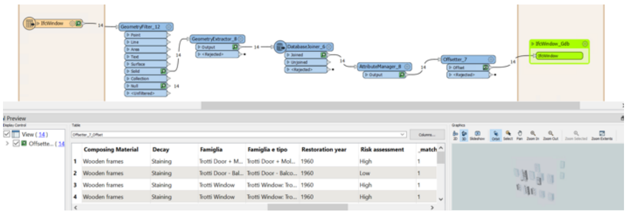


Figure 5. IfcMember and IfcBeam IFC-to-shapefile conversion.

This second part of the workflow was entirely carried by three fundamental players (i) the *DatabaseJoiner* transformer that joined the external Revit DB with the IFC one, through (ii) the GlobalID, that had been stored in each IFC single element, (iii) by the IfcGUID contained in each of the Revit schedules. The combination of the three allowed to manage the Revit elements, their attributes (namely, their shared parameters), and consequently, the geometric elements belonging to the IfcClasses. In this way, the Level of Information (LOI) of the Revit model was entirely retrieved though the use of the Revit schedules .txt (text) files, each schedule exploiting one of the Categories of the Revit project, for a total of 7 schedules, that resulted in a conversion of seven .gdb files, referring to the IFC 2X3 Classification supertypes, where (i) IfcDoor was converted in IfcDoor.gdb (Revit mapped category: Door, Italian: Porte); (ii) IfcRoof in IfcRoof.gdb (Revit mapped category: Roof, Italian: Tetti); (iii) IfcSlab in IfcSlab.gdb (Revit mapped category: Floors, Italian: Pavimenti); (iv) IfcWall, IfcWall StandardCase in IfcWall (Revit mapped category: Walls, Italian: Muri); (v) IfcWindow in IfcWindow.gdb (Revit mapped category: Windows, Italian: Finestre); (vi) IfcMember, IfcBeam in IfcBeam.gdb (Revit mapped category: Structural Framing, Italian: Telaio strutturale); (vii) IfcStair, IfcStairFlight in IfcStair.gdb (Revit mapped category: Stairs, Italian: Scale).



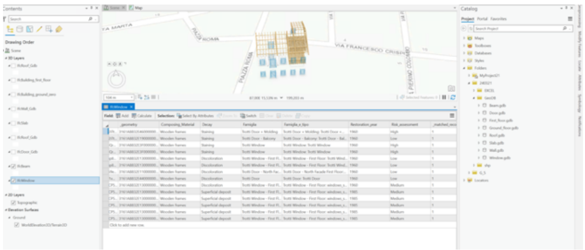
	Composing Material	Decay	Famiglia	Famiglia e tipo	Restoration year	Risk assessment	_match
1	Wooden frames	Staining	Trotti Door + M...	Trotti Door + Mol...	1960	High	1
2	Wooden frames	Staining	Trotti Door - Bal...	Trotti Door - Balco...	1960	Low	1
3	Wooden frames	Staining	Trotti Window	Trotti Window: Tro...	1960	High	1
4	Wooden frames	Staining	Trotti Window	Trotti Window: Tro...	1960	High	1

Figure 6. IfcWindow IFC-to-shapefile conversion, with semantic and geometric retrieving.

After the Join went successfully, the *AttributeManager* transformer was the one used to properly manage the retrieved attributes and easily allowed to add another column to each of the 7 .gdb, which aimed at storing the Revit classification category (in Italian), fundamental in the second part of the study, when in GIS environment, to facilitate the management of the HBIM through the Python interface. Lastly, the *Offsetter* transformer was to locate the elements in the space within the z coordinate (Figure 6). After running the full conversion algorithm in the FME Workbench, the elements of each class were opened in the FME Data Inspector extension, to check if both the geometric conversion and the semantic transfer were performed correctly.

4.2.4. Methodology test on ArcGis platform

After achieving a proper conversion for each one of the IfcClasses, the workflow switched to the ArcGIS Pro environment, where the first thing was to open a new project choosing the *Local Scene*; the local scene is used to display data that is spatially referenced in a projected coordinate system; local scenes can host projects from city to building scale, in this specific case it suited to the Historical Building scale. Then, after opening the new local scene, a connection to the specific folder containing the .gdb conversions was established through the ArcGIS Pro *Catalog* pane. Afterwards, to be able to perform scripts from the PyCharm interface, there was the need to perform a *FeatureClassToFeatureClass*, to convert each of the .gdb files into a project geodatabase file. The result of the conversion can be seen in the Fig.7, where the process satisfied the requirement of (i) keeping the information of the Revit model by not losing the LOI (Level of Information) during the conversion and by so, achieving the goal of having an attribute table that could be effective for an historical building condition assessment, easily accessible and prone to be implemented and consulted by multiple actors. The model is here georeferenced by its origin (Revit survey point), mapped in the IfcSite (Long., Lat)., then transformed into (E, N). A final transformation in FME is used to correctly reference the building with a roto-translation.



_geometry	Composing_Material	Decay	Famiglia	Famiglia_e_tipo	Restoration_year	Risk_assessment	_matched_reco
[illegible]	Wooden frames	Staining	Trotti Door + Molding	Trotti Door + Molding: Trotti Door +...	1960	High	1
[illegible]	Wooden frames	Staining	Trotti Door - Balcony	Trotti Door - Balcony: Trotti Door - Bal...	1960	Low	1
[illegible]	Wooden frames	Staining	Trotti Window	Trotti Window: Trotti Window	1960	High	1
[illegible]	Wooden frames	Staining	Trotti Window	Trotti Window: Trotti Window	1960	High	1
[illegible]	Wooden frames	Discoloration	Trotti Window - First Fl...	Trotti Window - First Floor: Trotti Wind...	1960	Low	1
[illegible]	Wooden frames	Discoloration	Trotti Window - First Fl...	Trotti Window - First Floor: Trotti Wind...	1960	Low	1
[illegible]	Wooden frames	Discoloration	Trotti Door - North Fac...	Trotti Door - North Facade First Floor:...	1960	Low	1
[illegible]	Wooden frames	Discoloration	Trotti Door	Trotti Door: Trotti Door	1960	Low	1
[illegible]	Wooden frames	Discoloration	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1960	Medium	1
[illegible]	Wooden frames	Superficial deposit	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1985	Medium	1
[illegible]	Wooden frames	Superficial deposit	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1985	Medium	1
[illegible]	Wooden frames	Superficial deposit	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1985	Medium	1
[illegible]	Wooden frames	Superficial deposit	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1985	Medium	1
[illegible]	Wooden frames	Superficial deposit	Trotti Window - First Fl...	Trotti Window - First Floor: windows_s...	1985	Medium	1

Figure 7. A sample of the IFC-to-shapefile converted elements opened in the ArcGis Pro environment

As the first original requirement was satisfied, the multipatch geometries could be then tested with both the (i) semantic and (ii) geometric geoprocessing, firstly semantic, querying the attribute table parameters, which resulted entirely queryable from a point of view of the organization of the geometries attributes; secondly, geometric by the means of the 3D Analyst geoprocessing, where the elements could deliver any kind of 3D intersection without running errors. This satisfies the second requirement of the methodology, which referred to overcome geometric losses and consequential problems deriving from 3D intersections. If the geometry had run errors, there should have been the need to use a workaround easy achievable by ArcGIS geoprocessing, involving both the *CheckGeometry* and *RepairGeometry* tools. Then, to properly deliver a back-end management of the elements belonging to the HBIM, a workflow was coded in the PyCharm environment, using the arcpy libraries to: (i) merge the feature classes, to "pack" the building elements , involving the *arcpy.Merge_management*; (ii) perform an Array selection of categories in order to simultaneously un-pack multiple feature classes, by the *arcpy.SelectLayerByAttribute_management,* (iii) copy in the .gdb by the *arcpy.CopyFeatures_management*, features from the input feature class or layer to a new feature class. All together, these elements contributed to achieve a slightly good interoperability within the HBIM-GIS framework in the ArcGIS environment, both Front and Back end.

4.2.5. Future challenges and issues

Concerning the here developed methodology, it is necessary to understand that the conversion was structured to follow the organization of the Revit libraries, where the IFC element was "mapped" from the Revit element. So, the conversion is partially relying on defining a good pre-mapping of the element in the Revit software, which entirely effects the way by which BIM objects are defined. This can represent a barrier for two reasons (i) the discipline barrier, as HBIM is itself an experimental research domain, which differences in knowledge definitions from other disciplines, and where the use of parametric software tools is an in-progress research for standardized methodologies to model the HBIM objects; so there is not a well-defined standardized vocabulary, which directly correlates with the second reason, (ii) the software barrier, as each software has a different representation method for the same geometry, in the specificities of its properties and relations. Furthermore, these aspects are to consider when coming to a formalization of the conversion, here affected with a bias for the Revit software (both in the modeling and in the object relations), which states its replicability in that software-oriented applications. Future challenges will be in the broad framework of a common methodology for modeling HBIM objects, coming with a standardized vocabulary to define their relations allowing a formalized approach in the applicative use.

5. CONCLUSIONS AND FUTURE APPLICATIONS

In conclusion, it is reasonable to say that the basic characteristics of the shapefile, e.g., the use of solid models and the separate management of geometry and semantics, make the conversion from IFC to shapefile a result that is quick to obtain and much flexible in the semantic transfer, giving to the shapefile a great potential in HBIM-GIS integration. In terms of achieving the IFC-shapefile geometric conversion, the single element conversion is more immediate since the shapefile can use the solid multipatch to represent models for all elements of the construction; in terms of semantic transfer, the attributes extracted from the 3D model and subsequently separately managed through relational databases, allow a fruitful retrieving of the 3D model information. Surely, future studies going in the direction of implementing standardized methods are whished for a more complete semantic transfer and more efficient geometry conversion. Further studies are needed to finalize better methodologies for organizing building information, especially building information in a context of spatial relationships, as a formal bottleneck of this methodology concerns the ineffectiveness of the shapefile in representing the spatial relationships between objects (as both the CityGML and the IFC do). Nevertheless, the workflow illustrated in the present paper allows to make very first considerations about the results obtained and the possible implementations that can derive from it. Moreover, further studies aiming at the accomplishment of defining a georeferencing standard for the BIM-GIS environment, could allow to achieve a complete geoprocessing of information between the 3D data and other GIS layers. For future applications we consider implementing the use of DBMS, where to organize the IFC file obtained from Revit in a relational database (DB) (e.g., Postgres/PostGIS), that could easily be accessed from any software package including ArcGIS. This will allow to structure a DB from which to directly query the elements of the HBIM, in respect of the use needed (e.g., this is a potential way to carry out a Condition Assessment of the HBIM). The data could also in this way be accessed by virtual web environments (e.g., Cesium), thus finalizing the consultation of the information and the real-time monitoring of the HBIM.

REFERENCES

Al-Saggaf, A. and Jrade, A., 2015, June: Benefits of integrating BIM and GIS in construction management and control. In *5th*

International Construction Specialty Conference of the Canadian Society for Civil Engineering (ICSC), 8-10.

Amirebrahimi, S., Rajabifard, A., Mendis, P. and Ngo, T., 2016: A BIM-GIS integration method in support of the assessment and 3D visualisation of flood damage to a building. *Journal of spatial science*, 61(2), 317-350. https://doi.org/10.1080/14498596.2016.1189365

Azhar, S., 2011: Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry. *Leadership and management in engineering*, 11(3), 241-252. https://doi.org/10.1061/(ASCE)LM.1943-5630.0000127

Bossaglia, R., Ferruzzi, A., Moioli, A., Villa, E., Zatti, M. S. 1990: Il Palazzo Trotti di Vimercate. Vimercate (MI).

Campanaro, D.M., Landeschi, G., Dell'Unto, N. and Touati, A.M.L., 2016: 3D GIS for cultural heritage restoration: A 'white box'workflow. *Journal of Cultural Heritage*, 18, 321-332. https://doi.org/10.1016/j.culher.2015.09.006

De Ruvo, V., 2019: Indagini sull'interoperabilità HBIM–GIS in un database europeo per la resilienza dei beni culturali= HBIM-GIS interoperability investigations in a european database for Cultural Heritage resilience (Doctoral dissertation, Politecnico di Torino).

Doumbouya, L., Guan, C.S., Gao, G. and Pan, Y., 2017: Application of BIM technology in design and construction: A case study of pharmaceutical industrial base of amino acid building project. *Engineering for Rural Development*, 16, 1495-02.

Ford, A., 2004: The visualisation of integrated 3D petroleum datasets in ArcGIS. In *Proceedings of 24th ESRI user conference*, 1-11.

Irizarry, J., Karan, E.P. and Jalaei, F., 2013: Integrating BIM and GIS to improve the visual monitoring of construction supply chain management. *Automation in construction*, 31, 241-254. https://doi.org/10.1016/j.autcon.2012.12.005

Isikdag, U., Zlatanova, S. and Underwood, J., 2013: A BIM-Oriented Model for supporting indoor navigation requirements. *Computers, Environment and Urban Systems*, 41, 112-123. https://doi.org/10.1016/j.compenvurbsys.2013.05.001

Jusuf, S.K., Mousseau, B., Godfroid, G. and Soh, J.H.V., 2017: Path to an integrated modelling between IFC and CityGML for neighborhood scale modelling. *Urban Science*, 1(3), 25. https://doi.org/10.3390/urbansci1030025

Li, X., Xu, J. and Zhang, Q., 2017: Research on construction schedule management based on BIM technology. *Procedia engineering*, 174, 657-667. https://doi.org/10.1016/j.proeng.2017.01.214

Longley, P.A., Goodchild, M.F., Maguire, D.J. and DW, R., 2005: Geographic Information Systems and Science. John Wiley.
Ma, Z., Ren, Y., 2017: Integrated application of BIM and GIS: an overview. *Procedia Engineering*, 196, 1072-1079. https://doi.org/10.1016/j.proeng.2017.08.064

Matrone, F., Colucci, E., De Ruvo, V., Lingua, A. and Spanò, A., 2019: HBIM IN A SEMANTIC 3D GIS DATABASE. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.

Penati, L., 1957: Vimercate. Raccolta di notizie storiche.

Saccucci, M., Cima, V., Grande, E., Imbimbo, M. and Pelliccio, A., 2019, November: The Knowledge Process in the Seismic Assessment of Masonry Building Aggregates–An Italian Case Study. In *International Conference on Critical Thinking in Sustainable Rehabilitation and Risk Management of the Built Environment*, 330-347. Springer, Cham.

Ramírez Eudave, R. and Ferreira, T.M., 2020: On the suitability of a unified GIS-BIM-HBIM framework for cataloguing and assessing vulnerability in Historic Urban Landscapes: a critical review. *International Journal of Geographical Information Science*, 1-31. https://doi.org/10.1080/13658816.2020.1844208

SAS, OPEN CASCADE 2019: https://www.opencascade.com/content/open-source-core-technology (30 January 2020).

Vacca, G., Quaquero, E., Pili, D. and Brandolini, M., 2018: GIS-HBIM integration for the management of historical buildings.

Volk, R., Stengel, J. and Schultmann, F., 2014: Building Information Modeling (BIM) for existing buildings—Literature review and future needs. *Automation in construction*, 38, 109-127. https://doi.org/10.1016/j.autcon.2013.10.023

Wang, H., Pan, Y. and Luo, X., 2019: Integration of BIM and GIS in sustainable built environment: A review and bibliometric analysis. *Automation in Construction*, 103, .41-52. https://doi.org/10.1016/j.autcon.2019.03.005

Zhu, J., Wright, G., Wang, J. and Wang, X., 2018: A critical review of the integration of geographic information system and building information modelling at the data level. *ISPRS International Journal of Geo-Information*, 7(2), 66. https://doi.org/10.3390/ijgi7020066

Zhu, J., Wang, X., Wang, P., Wu, Z. and Kim, M.J., 2019: Integration of BIM and GIS: Geometry from IFC to shapefile using open-source technology. *Automation in Construction*, 102, 105-119. https://doi.org/10.1016/j.autcon.2019.02.014

Zhu, J., & Wu, P. (in press): A systematic state-of-the-art review on BIM/GIS data integration from the perspective of information flow. *Automation in Construction*