POLITECNICO DI TORINO
Repository ISTITUZIONALE

IFC Classification for FOSS HBIM: Open Issues and a Schema Proposal for Cultural Heritage Assets

(Article begins on next page)

01 September 2025

IFC Classification for FOSS HBIM: Open Issues and a Schema Proposal for Cultural Heritage Assets

Filippo Diara *[ID] **and Fulvio Rinaudo**[ID]

Politecnico di Torino—DAD, Viale P.A. Mattioli, 39, 10125 Torino, Italy; fulvio.rinaudo@polito.it
* Correspondence: filippo.diara@polito.it

Abstract: The IFC (Industry Foundation Classes) open format has been developed by BuildingSMART and regularized through ISO standards. It has been implemented into a BIM (Building Information Modeling) informative system for the AEC industry (Architecture Engineering and Construction). The IFC format has changed interoperability processes concerning architectural and technical entities in a semantic way. However, because this standard open format was specifically designed for the modern AEC industry, it may not cater to the demands of cultural heritage assets. Since IFC classification is fundamental for informative systems, it should become a standard also concerning heritage assets, even if nowadays there is no regularized IFC classification for historical existing buildings. Specific cultural heritage peculiarities therefore need semantic classification based on historical asset families. For this reason, this work is based on a proposal and experimental IFC classification implemented inside an HBIM open source software (FreeCAD), whereby limitations of IFC standards can be overcome thanks to the freedom of access to libraries and codes. Moreover, this work is based on IFC objects management outside the platform for interoperability purposes.

Keywords: BIM; cultural heritage; DBMS; FOSS; HBIM; IFC; interoperability

---

1. Introduction

1.1. BIM and IFC Standard

The Building Information Modeling (BIM) methodology is changing the traditional ways of producing and designing documentation and analysis of new installations (for AEC industry) as well as existing buildings. BIM is changing the 3D models' meaning: it allows us to establish semantic and ontological relations between digital models and related information [1,2], composing an informative model concerning engineering and construction [3,4]. According to the National Building Information Model Standard [5] definition, BIM is referred to as a digital and complete representation of a building or architecture. BIM encloses attributes and technical features of digital parametric models. For this reason, it has been designed as a smart platform for knowledge sharing, easing construction and maintenance processes within the building life-cycle. The combination of metric information with important historical data could be therefore defined as a custom ontology concerning case study, in order to ease the information and knowledge description, comprehension and sharing processes [2].

Through the utilization of integrated databases, BIM solutions are able to collect and manage information concerning buildings and, for this reason, query processes become fundamental in order to investigate and validate the semantic dimension [6]. These functionalities encourage collaborations between different professionals through the utilization of open standards and formats in a multidisciplinary approach. As far as open standards and protocols are concerned, the IFC format (Industry Foundation Classes) has a fundamental role concerning data exchange and the interoperability of informative BIM platforms.

IFC is referred to as an open format, developed by buildingSMART [7,8]. Being widely adopted in the AEC industry and BIM applications, it has become a standard open format regularized by the International Organization for Standardization (ISO). IFC's main peculiarities are defined by the standards ISO 16739-1:2018 and ISO/TC 184/SC 4 [9,10]. These standards establish the dictionary terms and the IFC schema composition: the header section, types, entities, functions. In detail, information about the schema is given by the header section; types and entities are referred to as data definitions; functions and rules concerning the constraints regularized on data definitions are developed [9–11]. For example, inside the IFC schema, single entities (like ifcWindow, ifcWall) are composed by language names; semantic definitions (entity definition—attribute definitions—formal propositions); inherited definitions from super-types (entity and attribute inheritance); geometry use definitions, including local placement of the entity, geometric and parametric representation (limited and stable guidelines designed for AEC industry).

The IFC classification is furthermore divided into several editions and releases [12]. From 1996 until now, buildingSMART has released different versions of IFC: nowadays, the official versions are the IFC2x Edition 3 Technical Corrigendum 1 (named IFC2x3 TC1 or IFC2x3) and the IFC4.1. IFC4.1 will become increasingly required, even if the IFC2x3 is still the principal baseline for applications. The IFC2x3 schema is based on MVD (Model View Definition). It has been designed for the purpose of exchanging protocols and requirements of BIM-based projects for the AEC industry [7–10].

For this reason, the IFC open format could be considered a dynamic database format, because it allows for the inclusion of different information referring to relational semantic objects including layers and different classes.

BIM informative models differ from a methodological as well as conceptual point of view from classical 3D reality-based models, approaching more and more the GIS (Geographic Information System) concept methodology, even if the BIM approach maintains the modeling and representation scale peculiarities. In fact, BIM platforms require dynamic, simplified and lightweight models and formats, because the main actor and the real big data should be the information inside every object [13].

1.2. HBIM and IFC: State of Art

Since HBIM (Historic Building Information Modeling) was introduced inside the cultural heritage research panorama [14,15], it has experienced enormous growth due to its possibilities concerning the documentation and analysis of existing historical buildings, basing the entire procedure on the scan-to-BIM workflow (from metric survey to BIM models). However, this workflow typology shows critical points concerning the modeling phase and the IFC classification. In fact, heritage assets are characterized by decorations, imperfections and features that experience and increase effort in survey and modeling steps [16]. At the same time, the obtained parametric models of heritage assets require semantic and typological dimension by classifying objects into architectural entities. For example, a polyhedron has to be assigned to a specific architectural typology or sub-division in order to be classified as wall, becoming an informative object including semantic information. These typological IFC classifications have been designed exclusively for the AEC industry. Currently, heritage assets do not have a specific and ad hoc classification system for their peculiarities and the possibility of integrating predefined new architectural families could require too much time [17]. Despite the benefits of the BIM methodology for cultural heritage (especially for monitoring and valorization processes), the lack of IFC classification is an ongoing issue, especially for the semantic dimension, which is fundamental even for heritage assets. However, there are some differences: IFC classification for the AEC industry, based on ISO standards, has precise semantic meanings; semantic classification of heritage assets is dynamic and it can often be based on subjective interpretation of heritage peculiarities. For this reason, HBIM projects should focus on fundamental information regarding heritage assets (especially historic data), avoiding a strict IFC semantic classification, of which a proper classification for heritage assets will be difficult to obtain.

For this reason, semantic data of historical buildings and heritage assets require secure informative platforms for sharing knowledge among project collaborators: for example, historical analyses of masonries (mostly performed by merging digital data with paper-based studies) are fundamental in order to comprehend the evolution of historical buildings as well as in order to carry out monitoring analyses or preventive actions.

2. IFC Customization and Proposal for Heritage Assets

Limitations and issues previously mentioned have led actual researchers to consider new questions for the purpose of finding a solution to this problem. Recently, the experimentation of FOSS (Free and Open Source Software) solutions has attracted great interest in the HBIM panorama of research [6,18–20], especially for the purpose of overcoming BIM software limitations as well as validating open source possibilities, from software to open libraries, from open formats to data exchange. The presented project focuses on IFC classification analysis and a new demonstrative and proposal schema for heritage assets by using and exploiting FreeCAD (FOSS HBIM platform) possibilities: indeed, by acting directly on the open and accessible source code of Python modules and libraries, it is possible to adapt specific instruments to specific needs concerning cultural heritage assets.

FreeCAD, like commercial BIM software (e.g., Autodesk Revit or Archicad), does not contain predefined IFC definitions related to historical buildings. In fact, FreeCAD developers have embedded principal architectural families inside the software by drawing upon buildingSMART standards. This standard classification, referred to as IFC2x3 and IFC4.1 schemas, has been designed for the AEC industry (architectural design and engineering projects) and does not include historical architectural families, descriptions and attributes. This is a critical point to be solved also as far as heritage assets modeling is concerned [21,22].

2.1. IFC Custom Classification

FreeCAD software (version 0.17 and 0.18) [23], being a CAD/BIM parametric modeler, includes by default IFC2x3 and IFC4.1 schema inside Python files related to ARCH workbench modules. However, despite this integration, FreeCAD requires the additional and mandatory *ifcOpenShell* open library [24] for the purpose of managing IFC objects (import and export). For this reason, this Python library has been implemented inside FreeCAD.

IFC classification is included inside Python files (.py) concerning principal architectural ARCH modules of FreeCAD: *ArchWall.py*; *ArchStructure.py*; *ArchComponent.py*; *ArchRoof.py*; *ArchWindow.py*. Graphically, the IFC classification is selectable from the drop-down menu of attributes and it is referred to as the "*ifcRole*" property on FreeCAD 0.17 and 0.18 and as the "*ifcType*" property on the FreeCAD 0.19 pre-release. By using FreeCAD version 0.17, the IFC classification is accessible and implementable by managing every Python file related to module scripts. For example, the *ifcRole* property of ARCH_WALL module can be implemented by adding custom roles inside the *ArchWall.py* file (Figures 1 and 2). The same strategy could be performed to modify the ARCH_STRUCTURE module by customizing the *ArchStructure.py* file and modifying the role values.

By using the latest stable version of FreeCAD(0.18), the IFC roles list is embedded into a single Python file referred to as *ArchComponent.py* (Figures 1 and 3). Inside this file, there are *IFC types* list, *Property types* list, *Possible roles for FreeCAD BIM objects*. By implementing semantic definitions of roles (and then the internal IFC schema adopted by FreeCAD developers), the IFC classification of FreeCAD can be enriched.

Both types of modifications (directly on source code of Python files) allow us to access graphically these implementations through the drop-down IFC menu, which contains default and modified classifications selectable as well as compatible with the next query analysis.
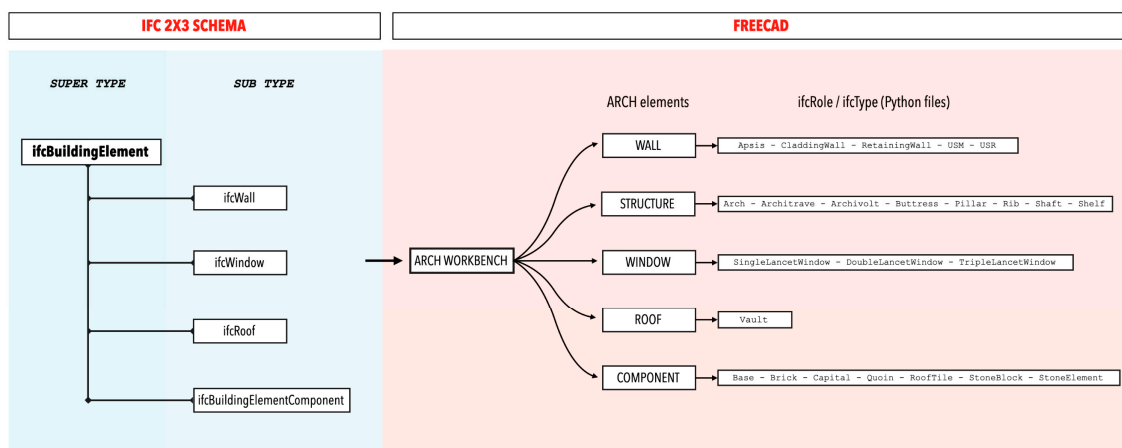
Figure 1. Proposal extension of *ifcRole/ifcType* related to Python files of ARCH elements inside FreeCAD.
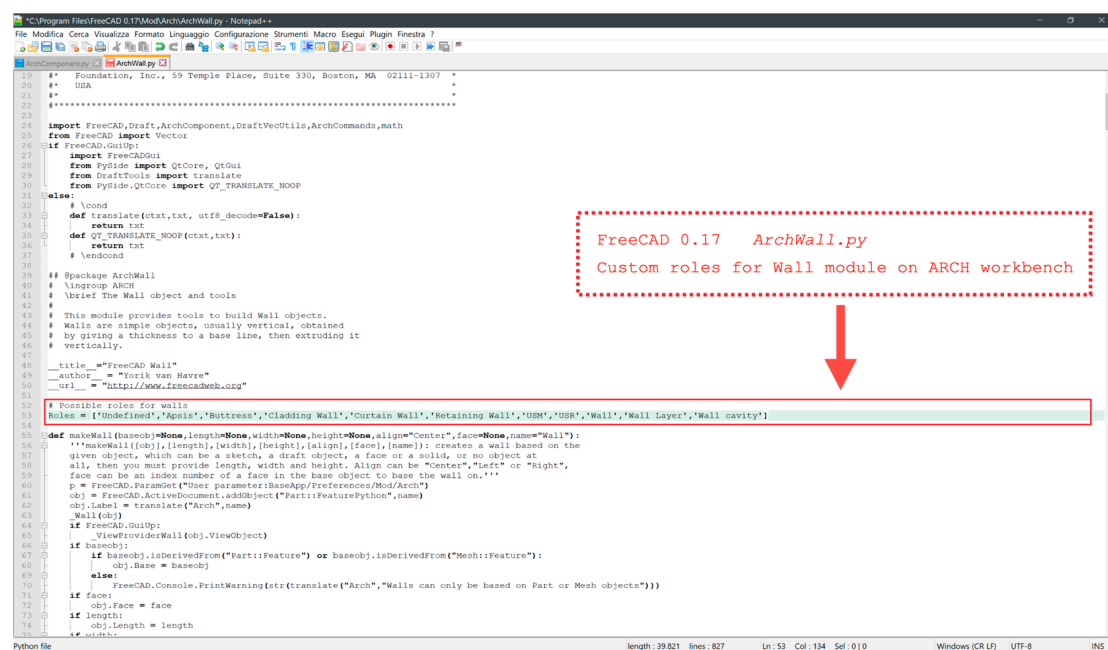

Figure 2. Access and modification of ArchWall.py Python file by using FreeCAD 0.17: highlighted in green are customized and possible roles for wall entity and related attributes below.

2.2. Schema Proposal for Heritage Assets

Definitions of historical architectural entities, as well as the parametric family classification, inside the BuildingSMART schemas and standards is therefore the main ongoing research field for HBIM purposes.

By accessing the Python source code of entities, the modification and implementation of FreeCAD IFC classification is nevertheless possible. The main idea has been the demonstrative extension of IFC entities by using particular historical architecture peculiarities derived from building archaeology purposes, even if they are not provided with the integration inside the IFC2x3 and IFC4.1 standard schema. Due to compatibility and interoperability issues regarding the lack of cultural heritage asset IFC standards, this kind of implementation is only related to the FreeCAD environment and it should be considered a proposal as well as a demonstrative modification of *roles* and *types* classification. As mentioned before, this classification is defined as "*ifcRole*" on version 0.17–0.18 and "*ifcType*" on 0.19 pre-release version by FreeCAD developers.

Figure 3. Access and modification of ArchComponent.py Python file by using FreeCAD 0.18: highlighted in green are customized and possible roles for BIM objects and related attributes below.

This proposed hierarchical schema of ARCH Workbench of FreeCAD has been designed for including custom entities (objects definitions) to existing default modules: they have been considered as roles of ARCH modules as ifcWall, ifcWindow, and so on. Exactly as ifcWall and other entities inherit features of ifcBuildingElement (super-type) on the official IFC2x3 schema, implemented roles/types of FreeCAD inherit the main characteristics of generic upper entities like wall, window, roof referring to ARCH elements.

IFC schema implementation has been designed for some entities referred to medieval historic architecture, medieval archaeology and building archaeology peculiarities, in order to ease the classification and documentation of heritage features through the FreeCAD HBIM platform.

The default classification has been implemented with different entities like Apsis—Arch—Architrave—Archivolt—Base—Brick—Buttress—Capital—Cladding Wall—Double-Lancet Window—Pillar—Quoin—Retaining Wall—Rib—Roof Tile—Shaft—Shelf—Single-Lancet Window—Stone Block—Stone Element—Triple-Lancet Window—Vault (Figure 1 and Table 1). These entities have been selected from medieval architecture elements which recur inside building archaeology documentation processes.

Table 1. Example of custom implementation referring to main elements of historic architecture and building archaeology. These definitions have been included on "*ifcRole*" menu of FreeCAD.

Wall	Structure	Roof	Window	Component
Apsis	Arch	Vault	Single-Lancet	Base
Cladding Wall	Architrave		Double-Lancet	Brick
Retaining Wall	Archivolt		Triple-Lancet	Capital
USM	Buttress			Quoin
USR	Pillar			Roof Tile
	Rib			Stone Block
	Shaft			Stone Element
	Shelf			

Of course, this is a preliminary list of historic architectural entities that could be implemented with other important elements. Moreover, inside this list, there are two entities with a double semantic value: indeed, due to the general definition *block* and *element*, the addition of material semantic information (in this case, *stone*) may be fundamental for documentation and analysis.

According to different versions of FreeCAD, this IFC implementation has been performed in two different ways. Firstly, by following the 0.17 version of FreeCAD, this custom list has been sub-divided among different Python files mostly concerning ARCH_WALL, ARCH_STRUCTURE, ARCH_ROOF, ARCH_WINDOW and ARCH_COMPONENT principal macro entities. (Table 1).

In FreeCAD version 0.18 (which has been used for practical demonstrations), the IFC implementation has been directly performed by editing the single *ArchComponent.py* Python file (implementing previous custom types) because it contains *IFC types* list, the *Property types* list and the *Possible roles for FreeCAD BIM objects*, differently from previous versions of FreeCAD.

Despite the original aim of IFC classification (for the AEC industry), Python files allow us to customize the "*ifcRole*" referring to FreeCAD macro elements and it has been performed regarding peculiarities of archaeological sites and building archaeology. The most important information derived from these disciplines is referred to as stratigraphy and its analysis is based on physical sequences among layers and units. Indeed, the stratigraphic method, derived from geological analyses and then from field archaeology, can be therefore applied in historical buildings (mostly medieval), and stratigraphic units (portions that compose the stratigraphy) can be analyzed and documented through the edges.

Stratigraphic units (US) can be mostly referred to as positive (accumulation of materials) and negative (removing activities) actions and can be also referred to in terms of different compositions: for example, in building archaeology, field principal differences are related to masonry units (USM), render/plaster units (USR) as well as architectural elements (EA). FreeCAD classification concerning the archaeological documentation method may include the classification of archaeological units as archaeological evidence, including all reference unit typologies [25]: masonry stratigraphic unit (USM), render/plaster stratigraphic unit (USR), architectural element (EA), architectural complex (CA), block of the building (CF), general elevation (PG), particular elevation (PP), horizontal structure (SO), functional unit (UF). These classifications might be included in FreeCAD 0.17 inside macro entities like ARCH_WALL, ARCH_STRUCTURE and ARCH_COMPONENT, from the biggest to the smallest unit.

In the FreeCAD platform, the classification referring to masonry and render/plaster units detectable in building archaeology contexts has been included. USM and USR classification has been designed as custom "*ifcRole*" of *ifcWall* entity, since these unit typologies are part of walls and masonries. This classification could also be developed by exploiting ifcMaterialLayerSets inside the IFC2x3 schema and by defining materials' layers and thicknesses (analyzing units in more detail). However, in this project, the main focus has been an early attribution of typological (and more generic) definitions to particular heritage assets via FreeCAD possibilities.

After the implementation inside the Python file of the FreeCAD folder root, modifications become available on the FreeCAD interface inside the drop-down menu referring to "*ifcRole*": attributes and properties of new classification refer to general and structural behaviors of the principal entities which they rely on. After this implementation, different experimentations have been carried out by using some examples concerning heritage assets: an ionic column, a rib vault and a medieval masonry. These patterns have been examined and documented in order to validate the proposed IFC classification.

An ionic column (Figure 4), consisting of a base, a shaft and a capital, can be therefore documented and semantically analyzed by using the FreeCAD custom HBIM platform. After the inclusion of ad hoc definitions, the components of the column can be correctly classified: the custom "*ifcRole*" *capital*, included inside the ARCH_COMPONENT module, can be assigned to the decorated capital of the column; *shaft* "*ifcRole*", considered as the role of *ifcColumn* (inside the ARCH_STRUCTURE module), can be assigned to the central vertical part of the column (shaft); *base* "*ifcRole*", included inside the ARCH_COMPONENT module, can be assigned to the lower support of the column (base).

By modifying and implementing the ARCH_ROOF module, other particular typologies of ceilings and roofs can be included, following the attributes and properties of main entities. A medieval vaulted

system (consisting of a rib vault) has been used as an example for semantic IFC classification inside the custom HBIM platform (Figures 5 and 6).
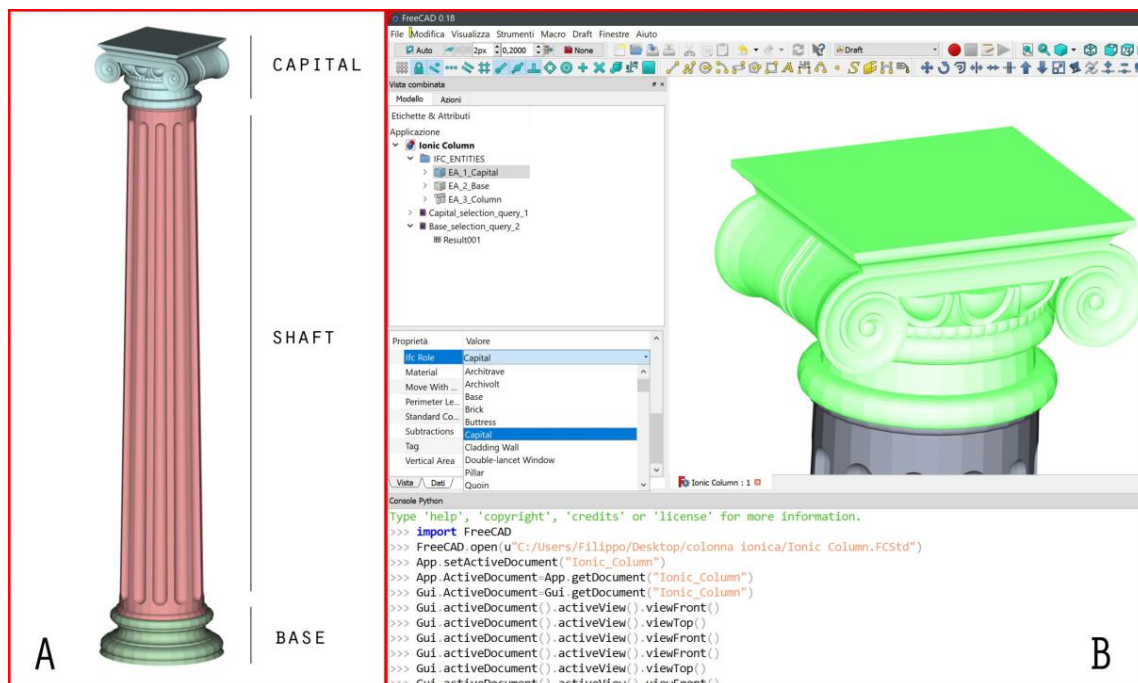

Figure 4. Example of ionic column. (**A**) Ionic column subdivisions: capital in blue; column shaft in red; base in green. (**B**) FreeCAD experimentation: IFC classification of the ionic capital by using custom roles available on the drop-down menu after the implementation.
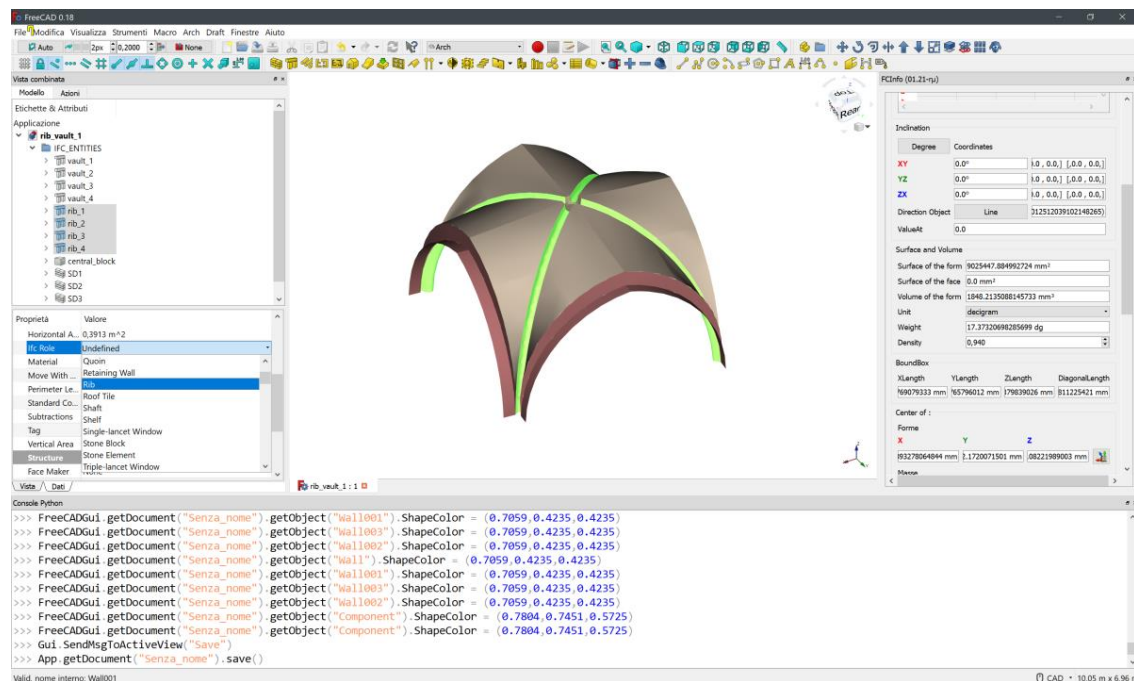

Figure 5. Example of a medieval rib vault: IFC classification of ribs by using custom roles available on the drop-down menu after the implementation.
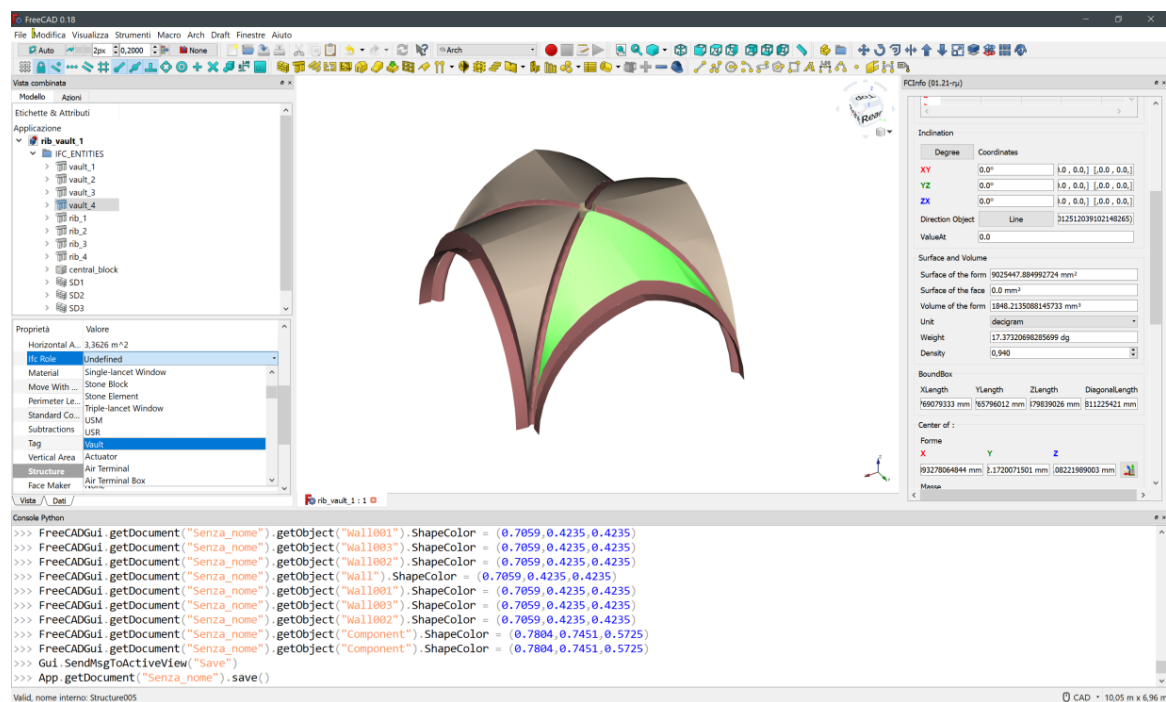
Figure 6. IFC classification of vaults by implementing ifcRoof entity by using *vault* as custom roles inside the default schema.

Moreover, documenting and analyzing a medieval rib vault it can be considered both as *roof* element and *structural* element due to its ribs supporting vaults; then, following this line of thought, the custom roles of *rib* and *vault* have been included inside the ARCH_ROOF and ARCH_STRUCTURE modules of the ARCH workbench. In this way, a rib vault can be semantically and structurally classified following its components and their behaviors.

The third example has been a medieval wall consisting of stone masonry, a single-lancet window contemporary with the main construction site and its plaster/render layer can be documented and classified by using custom definitions implemented in default Python files of FreeCAD (Figures 7–11). Looking at the building stratigraphy, the single-lancet window, being in the same construction temporal phase with the principal stone masonry, could be inscribed into the principal stone masonry and then become part of the masonry (ifcWall). However, it must be classified as an architectural element (EA) and then it has to be assigned to an IFC custom family: in this case, the already implemented definition *single-lancet window* has been used as a role of ARCH_WINDOW in FreeCAD 0.17 and as a custom role of the STRUCTURE entity inside the generic BIM objects list of FreeCAD 0.18. The principal masonry unit as well as render/plaster unit could be classified by using the already implemented USM and USR definitions, designed as roles of the ifcWall entity, being fundamental components of the medieval wall.

The here presented conceptual medieval stone wall, which includes also a single-lancet window and a plaster/render layer, is undoubtedly a particular example. It allows different interpretations that can be extracted from the stratigraphic analysis, depending especially on surveyor interpretation (subjective analysis). This analysis may not be the only one that can be extracted from stratigraphy.

Archaeological analyses, like most humanities studies, are essentially based on different kinds of data, in addition to the information extractable from the surveyed context: from paper-based analyses to 2D drawings, from archive research to alphanumerical diagrams. These collected data are essentially used in order to make historical interpretations or to reconstruct an historical event (based on archaeological evidence). By definition, interpretation has a strong subjective factor and the hypothetic IFC classification is directly proportional to subjective analyses carried out by archaeologists or historians. For this reason, IFC definitions for historic architecture elements and stratigraphic units detectable on masonries could assume different keys of reading.
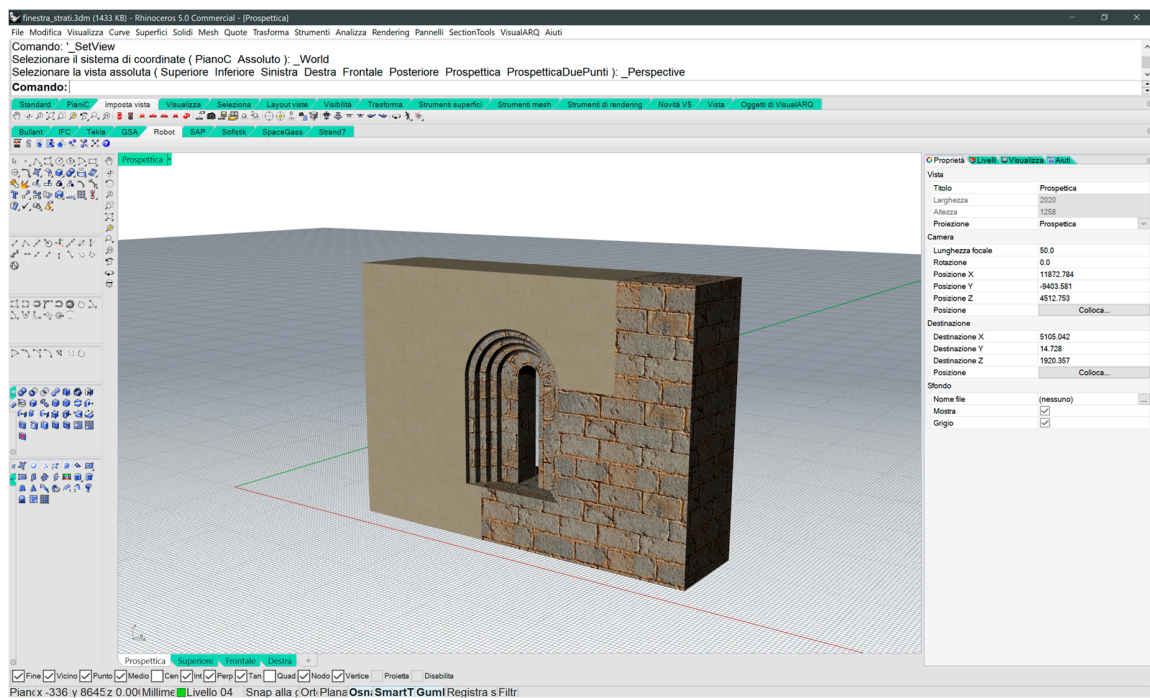
Figure 7. Simplified and conceptual 3D model (NURBS) of a medieval stone wall including plaster unit and a single-lancet window.
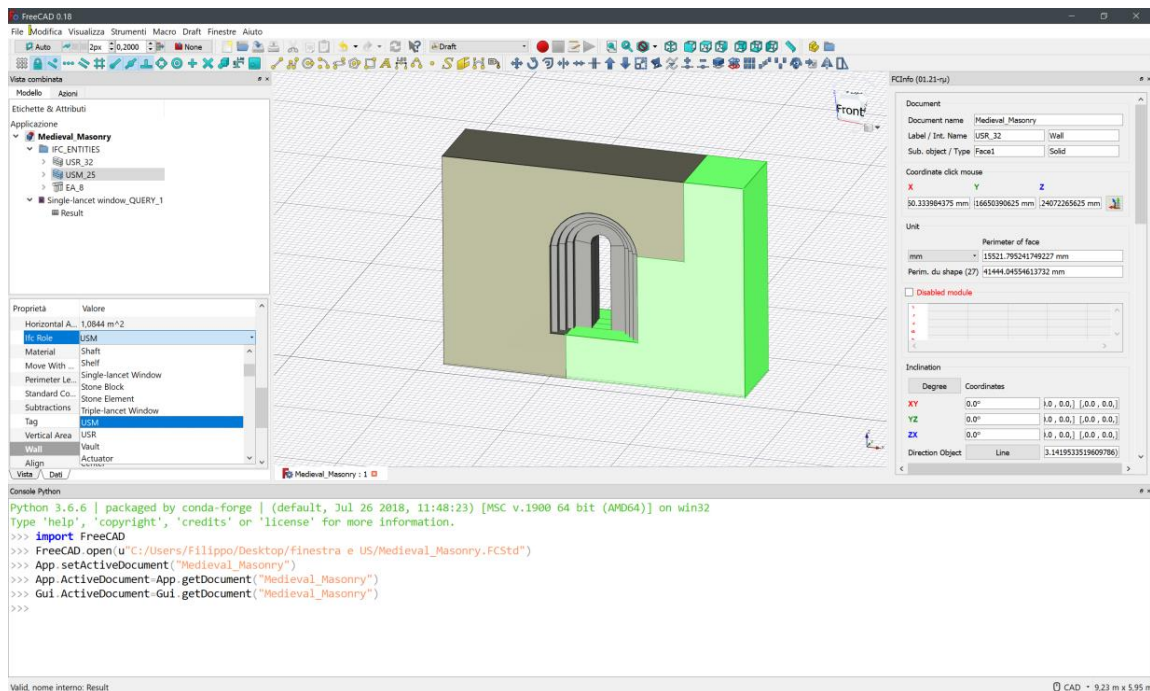

Figure 8. Example of a medieval stone wall: IFC classification of stone masonry as stratigraphic unit (USM on the right of single-lancet window—see also Figure 7) available on the drop-down menu after the implementation of ifcWall entity and ArchComponent.py.
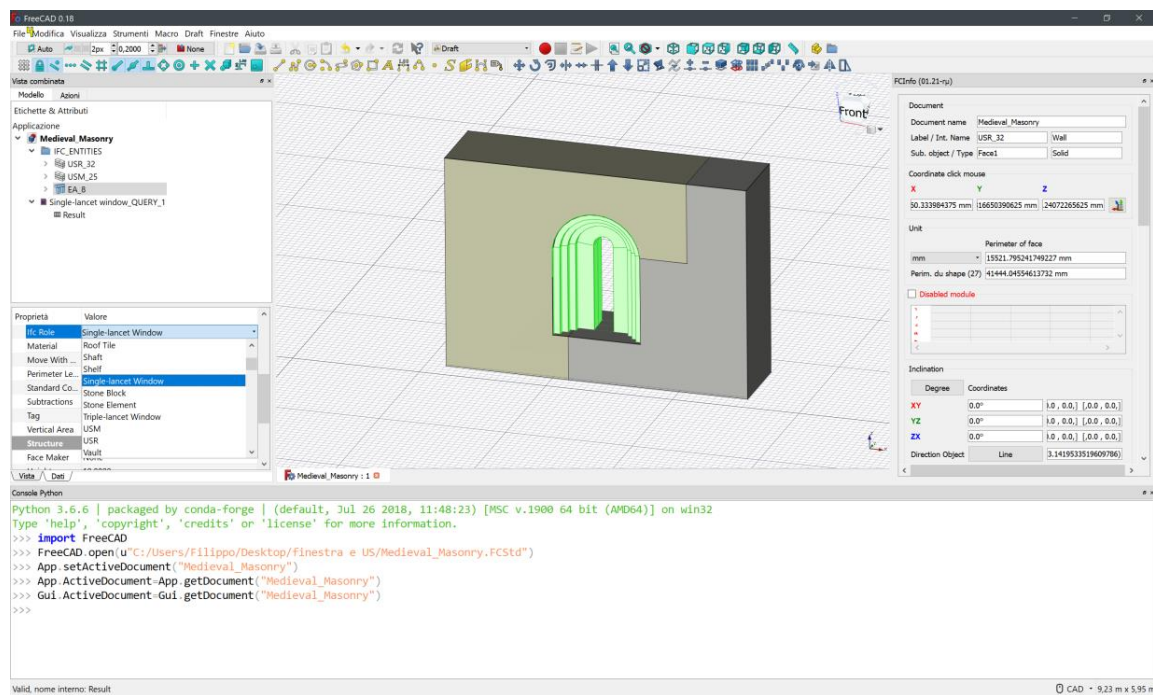
Figure 9. Example of a medieval stone wall: IFC classification of single-lancet window available on the drop-down menu after the implementation of ifcWindow entity and ArchComponent.py.
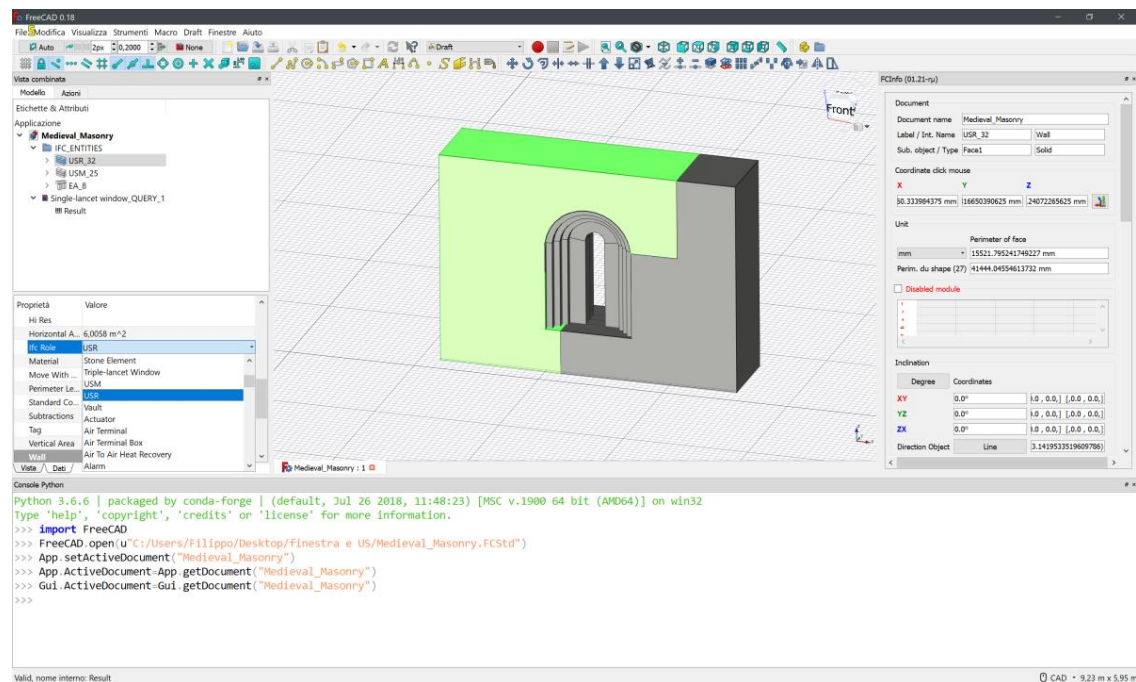

Figure 10. Example of a medieval stone wall: IFC classification of plaster layer on masonry as render stratigraphic unit (USR on the left of single-lancet window—see also Figure 7) available on the drop-down menu after the implementation of ifcWall entity and ArchComponent.py.

The single-lancet window example, according to stratigraphic evidence, can take a dual reading: contemporary and not contemporary to the stone masonry. Reflecting on this archaeological interpretation, the IFC hypothetical classification can change depending on this interpretation: a single-lancet window contemporary to a stone masonry could be therefore classified as the stone blocks' continuation of the masonry and then classifiable both as ifcWALL and ifcWINDOW due

to architectural element classification (as proposed in the example); a single-lancet window not contemporary to stone masonry (e.g., subsequently obtained by cutting the masonry) could be classified as a separate element and then could be used the ifcWINDOW or ifcOPENING_ELEMENT definitions. Then, the object classification of stratigraphic layers and units strongly depends on surveyor analysis.

Unlike engineering, constructions and technical installations, humanities and archaeology are not exact sciences because they rely on human interpretation (subjective) and these sciences do not depend on calculations and international standards. For this reason, an extended IFC classification for building archaeology (as well as semantic and chronological interpretation) could be correct for the archaeologist who carried out the analysis and wrong for other archaeologists (who give a different interpretation). This is not an issue for humanities studies, even if representation methods should be shared and defined regarding BIM purposes.

In this regard, FreeCAD software and its modified IFC classification proved to be dynamic and flexible for further changes and updates concerning semantic data, including definitions for historical architectural elements and archaeological stratigraphy.
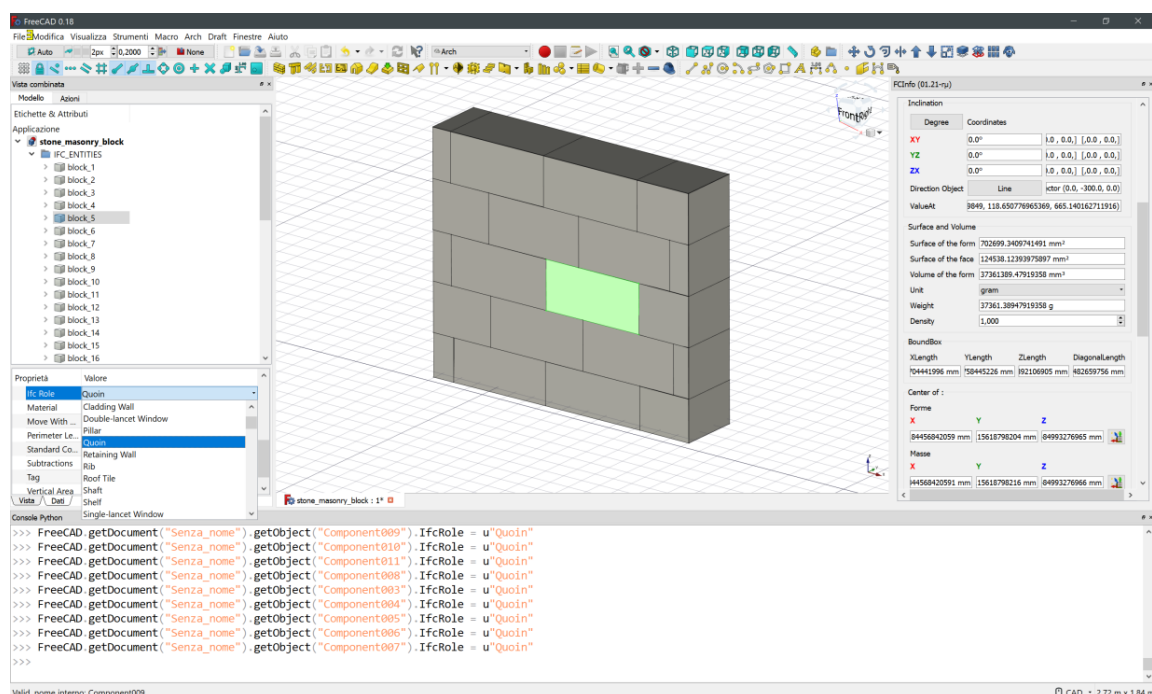

Figure 11. Detail of a medieval stone wall: possible IFC classification of stone quoins through the implementation of FreeCAD ArchComponent.py and managed as construction blocks (components).

3. Querying New IFC Entities

Query processes are a fundamental requirement for HBIM purposes to analyze and validate information concerning the case study, also for further investigations.

Inside the FreeCAD HBIM platform, query processes have been carried out by using the additional *Reporting Workbench* (easily downloadable through the "addons" window inside the software) due to its features regarding the design and management of custom SQL (Structured Query Language) queries and the related data. SQL language is used to communicate with common databases. It is the standard language for relational database management systems (RDMS). Moreover, the Express Query Language (EQL) is an additional and generic SQL-like language that allows the user to perform queries (using SQL statements) on data available in STEP format. IFC has been developed based on STEP and EXPRESS; therefore, the EQL language can be applied to IFC objects [26–28].

To ease investigation of the new custom IFC classification—which has been revealed to be fully compatible with SQL investigations through *Reporting Workbench*—custom SQL queries based on SELECT syntax have been designed. SELECT is one of the most important SQL statements to extract data from databases (in this case, the internal database of FreeCAD).

SQL investigations that could be performed by using the Python console as well as statements and datasheets of Reporting Workbench have been developed to query new IFC entities. Different SQL queries have been designed for the examples previously analyzed: the ionic column; the medieval rib vault; the medieval stone wall. Going further, the Python console query method has been chosen to select specific roles of BIM objects through their project label; datasheet-based queries have been set in order to select BIM objects belonging to the same IFC family, reporting object label and the related short description on datasheets.

Figure 12 shows an example of SQL query based on SELECT syntax and Python console method in order to select *Vault* IFC role from project objects (first two rows are considered for the purpose of evoking the already installed SQL parser module inside FreeCAD):

from sql import freecad_sql_parser
sql_parser = freecad_sql_parser.newParser()
select_something = sql_parser.parse("Select Label from document Where IfcRole = 'Vault'")
select_something.execute()

This query, directly written on the Python console, generates a green text string output visible below the query text string, including (through square brackets) the resulting object labels. At the same time, this SQL query can be used to investigate other IFC roles, only changing *ifcRole* with the other example roles concerning the examples. As far as the example of the ionic column and capital is concerned, Figures 13 and 14 show the SQL statement of the datasheet-based query for the purpose of selecting object label and description of *Capital* IFC object:

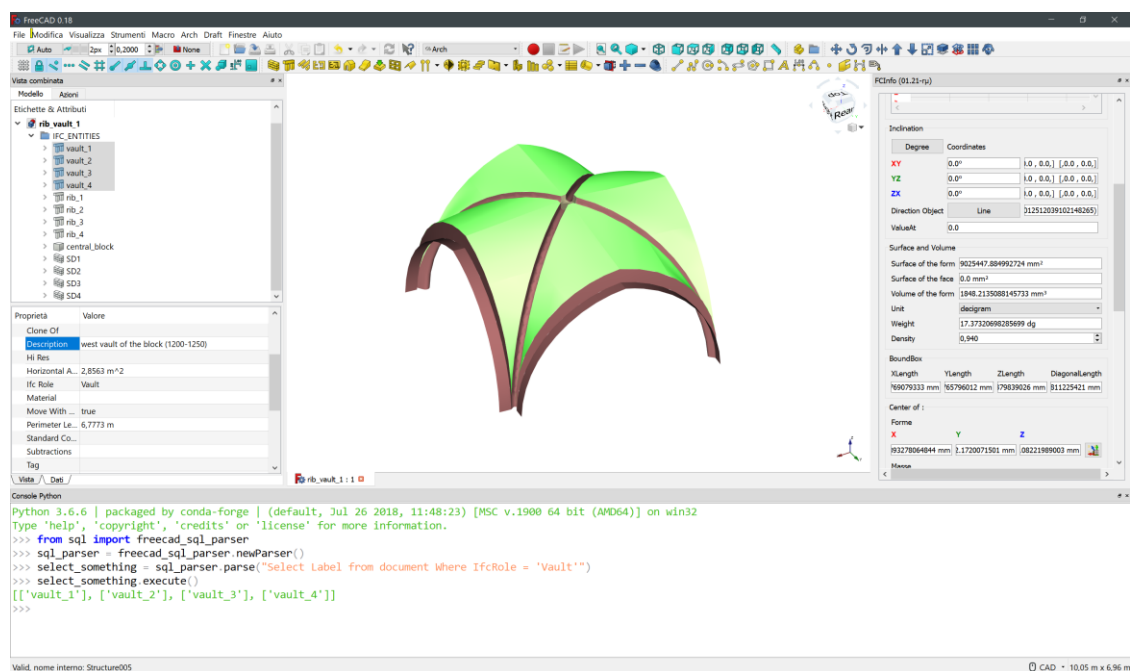Select Label, Description from document Where IfcRole = 'Capital'

Figure 12. SQL query performed on Python console in order to investigate *Vault* IFC custom role: results in green text inside square brackets after the SQL query syntax.
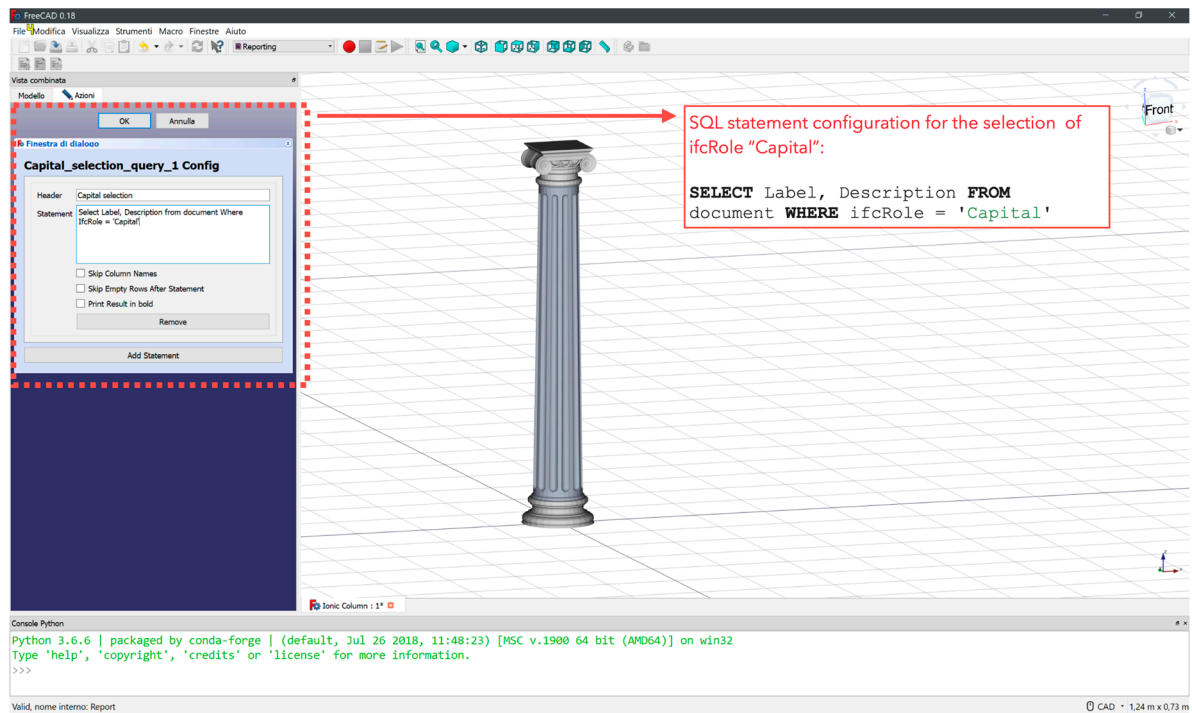
Figure 13. SQL query performed by using predefined statements and datasheet report in order to investigate *Capital* IFC custom role and its description: on the left side of FreeCAD is the SQL statement preparation for custom query.
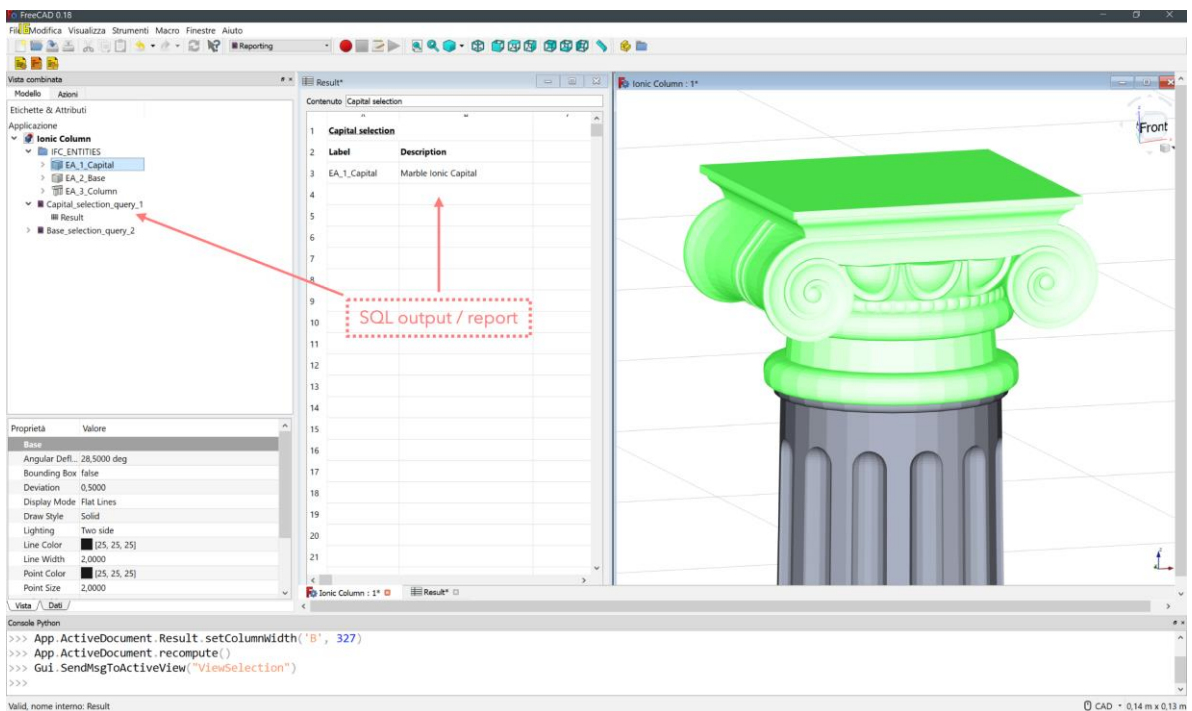

Figure 14. SQL query in order to investigate *Capital* IFC custom role: on the left side are the saved *capital_selection_query_1* and the related datasheet report with the query result.

This SQL query method, easy to manage thanks to ad hoc query integrated into the project tree inside FreeCAD, generates a datasheet report (exportable in CSV format) concerning the result of the query investigation (as described in the next paragraph). As in the previous method, this text string—referring to a selection of object labels and description of *Capital* IFC object—can be therefore

used and modified for the other analyzed example, simply changing "*Capital*" with specific roles. More sample queries have been performed as regards rib vaults investigation (Figures 15 and 16) as well as the medieval masonry, especially the included single-lancet window (Figures 17 and 18).
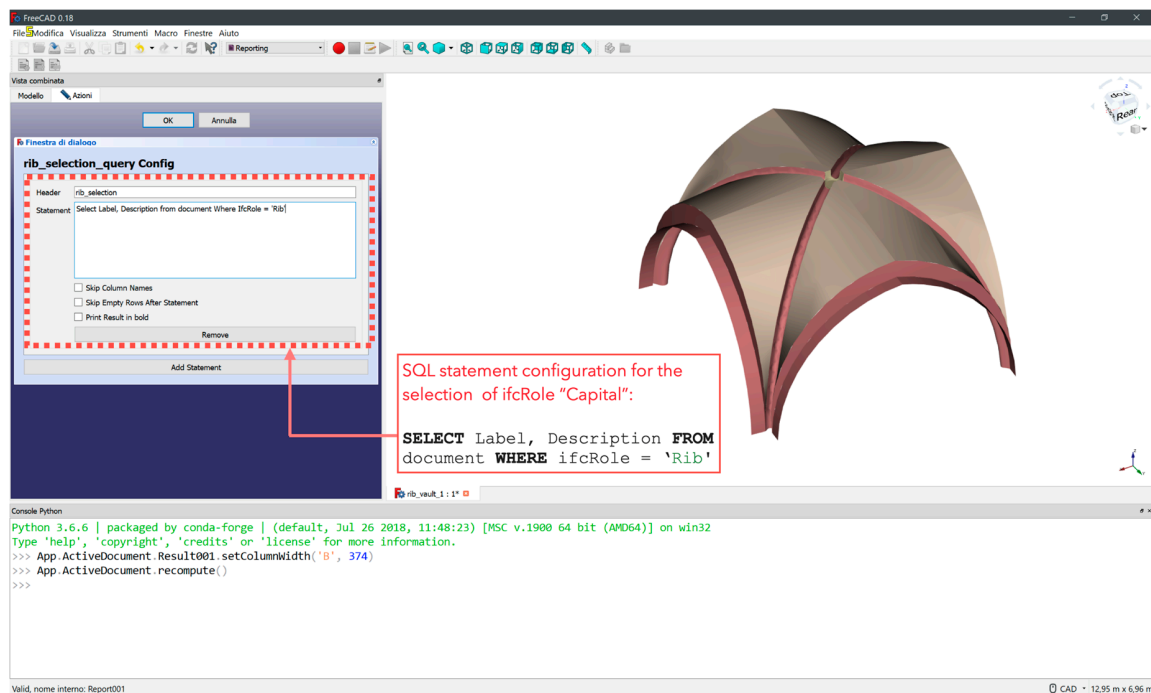

Figure 15. SQL query performed by using predefined statements and datasheet report in order to investigate *Rib* IFC custom role and its description: on the left side of FreeCAD is the SQL SELECT statement preparation.
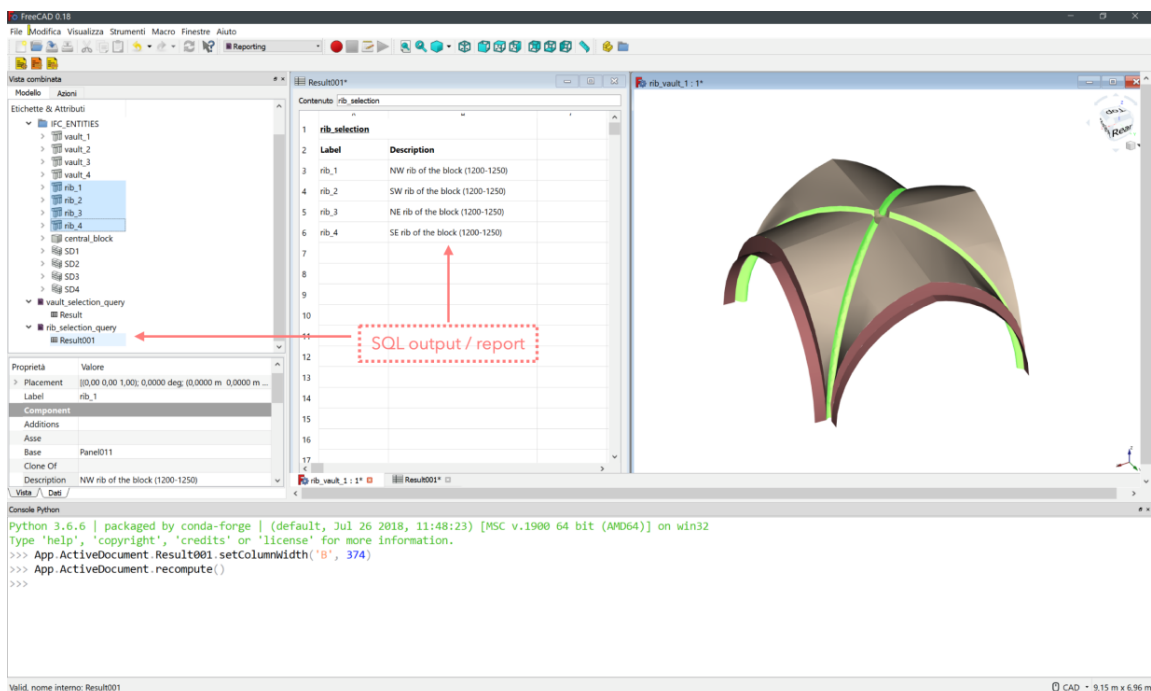

Figure 16. SQL query in order to investigate *Rib* IFC custom role referred to rib vault and their description: on the left side are the saved *rib_selection_query* and the related datasheet report with the query result.
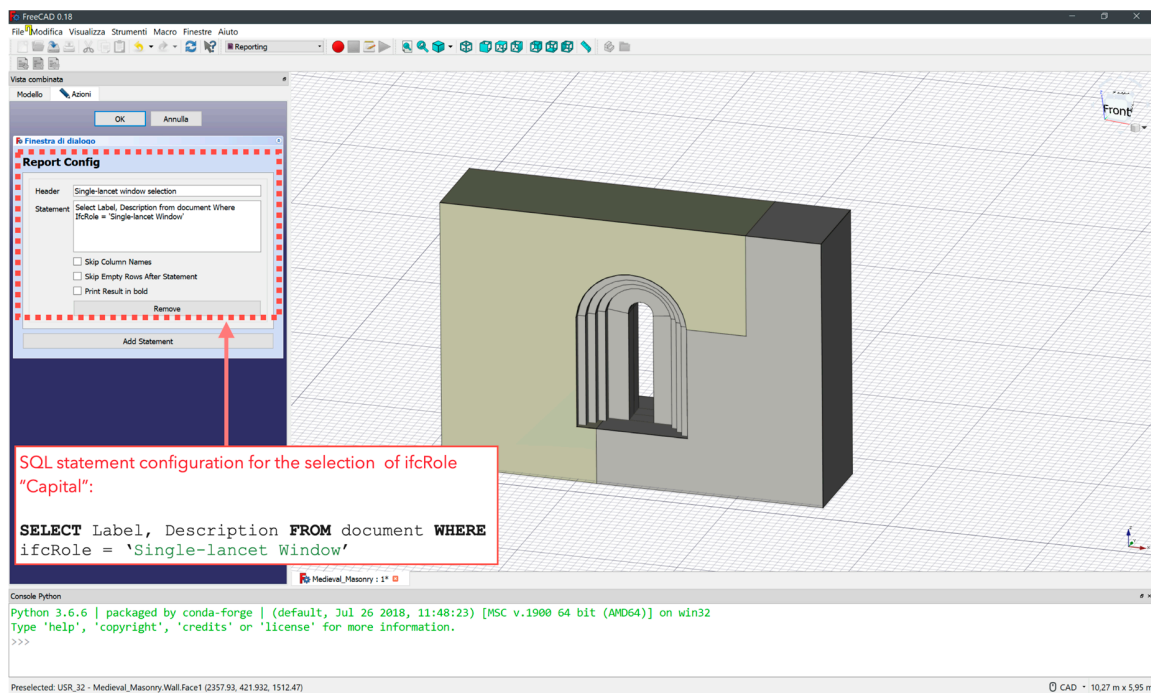
Figure 17. SQL query performed by using predefined statements and datasheet report in order to investigate *single-lancet window* IFC custom role and its description: on the left side is the statement preparation.
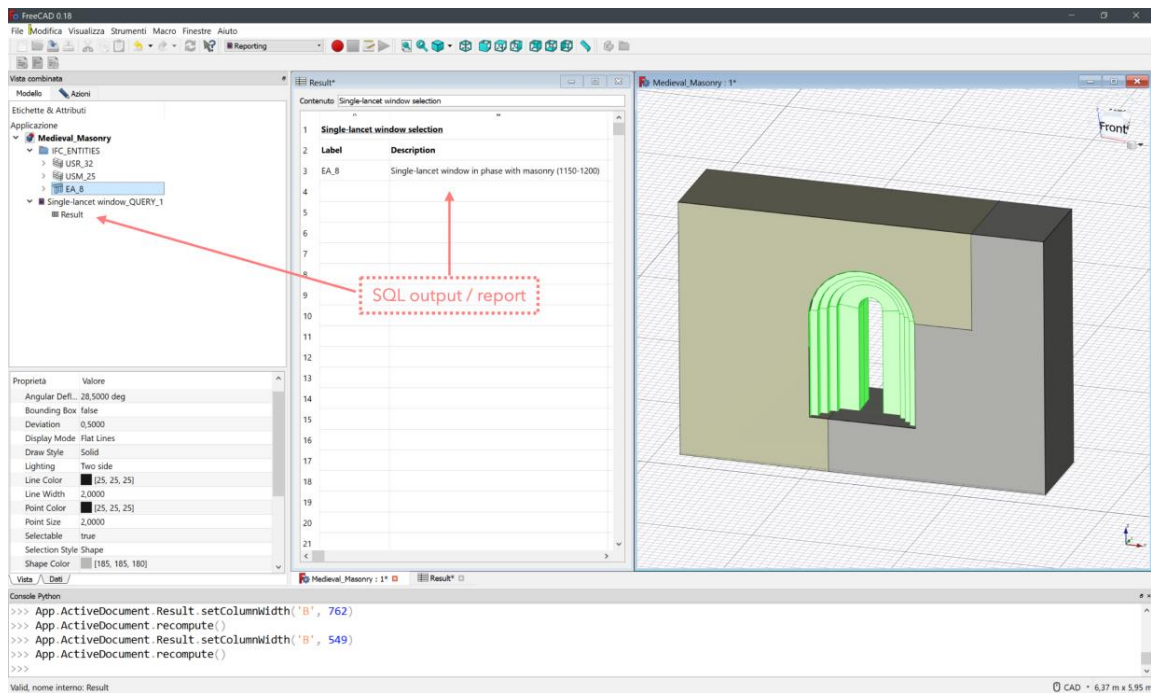

Figure 18. SQL query in order to investigate *single-lancet window* IFC custom role and its description, referring to the medieval wall: on the left side are the saved *single-lancet window_query_1* and the related datasheet report with the query result.

SQL queries based on SELECT syntax can be also used for other kinds of investigations (replacing, for instance, *ifcRole* string from previous queries)—for example, to identify object materials, identification codes based on FreeCAD *standard code*, particular text-based tags concerning specific objects, etc. SELECT queries as well as relational queries play a fundamental role in deep analysis and

semantic knowledge of BIM objects and this is why actual BIM platforms, in addition to representing fundamental solutions concerning modeling and managing phases, shall be also considered safety collectors of knowledge regarding parametric objects.

4. Issues and Possible Solutions: Classification, Interoperability and Data Exchange

The described proposed classification concerning specific cultural heritage assets presents different methodological issues.

Despite this solution guaranteeing a fitting classification for some heritage peculiarities, it presents different drawbacks because it does not follow ISO standards. Moreover, this customized IFC classification is not recognized on other BIM platforms (e.g., Autodesk Revit) and management software based on these rules. Indeed, this demonstrative and proposed classification, due to the lack of IFC definition framework for cultural heritage assets, can be used only for local usage in the FreeCAD environment. FreeCAD can be adapted to perform queries on custom classification, but without exchanging "*ifcRole*" values and data on external servers, viewers and other management software for BIM objects. This important semantic and methodological issue also results in interoperability issues concerning the multidisciplinary approach by using different BIM software as well as management software. This is one of the "open issues" mentioned in the title which unfortunately remain, because of IFC ISO standards being exclusively for the AEC industry.

Knowing these limits, the proposed classification has been performed especially in order to prove the flexibility and dynamism of FreeCAD and FOSS solutions as far as the IFC custom definition is concerned: this local proposal is fully compatible with other BIM projects carried out with FreeCAD, even without modifying the original IFC schema. In this way, an HBIM project developer can share and exchange semantic data and ad hoc IFC classification within other FreeCAD projects.

The FreeCAD environment has revealed a complete solution concerning the management of BIM objects: from object surface parametrization [29–31] to the inclusion of ad hoc libraries easily adaptable to cultural heritage assets; from IFC type classification via Python source modification to integrated DBMS and ad hoc SQL queries designed for the previous ad hoc IFC classification (Figure 19).

However, IFC data exchange can be performed by respecting ISO standards and regulations. Although it is referred to as a generic classification, it can be carried out based on existing IFC families: in this case, entities such as USM (masonry stratigraphic unit) can be classified as a generic ifcWall (sub-type of ifcBuildingElement on the IFC2x3 schema); medieval single or double-lancet windows could be defined as ifcWindow; medieval vaulted systems could be simplified in ifcRoof. This solution would therefore be optimal for interoperability and data exchange among project collaborators through different software used, even if there is a strong semantic simplification and interpretation (fundamental aspect for heritage assets).

In order to solve interoperability and data exchange issues, different solutions could be developed to fix custom IFC classification vulnerabilities. The first possible solution is IFC File Analyzer, which is a software that generates CSV (comma-separated values spreadsheet) or XLSX (Excel file format for datasheets) files from IFC file [30]. Through this operation, all the information concerning the IFC file, including building elements, roles, materials, geometry, relationship and other properties, can be extracted (Figures 19 and 20). This solution could be reliable because, after the extraction process, CSV and XLSX files containing data from IFC objects can be stored and managed inside a relational or spatial DBMS [31], fundamental for the purpose of sharing information and knowledge.

CSV files and datasheets are essential to ease the comprehension of the IFC structure referring to single entities and objects. Then, as mentioned before, users who manage IFC files through the software analyzer can learn about the objects hierarchy, inspecting properties and relationships between the entities, even without the use of a BIM software. This is a fundamental feature of IFC File Analyzer, because through the generated CSV or XLSX datasheets, the knowledge dissemination of information and then data exchange can be guaranteed both for BIM managers and other researchers, ensuring

data preservation and dissemination in different types of DBMS. In this way, the quality of essential data is preserved.

The integrated database of FreeCAD and other external databases may communicate, becoming essential in order to collect, manage and spread a large amount of information concerning IFC custom objects. This cooperation would allow us to solve the already mentioned issues related to interoperability between IFC applications.
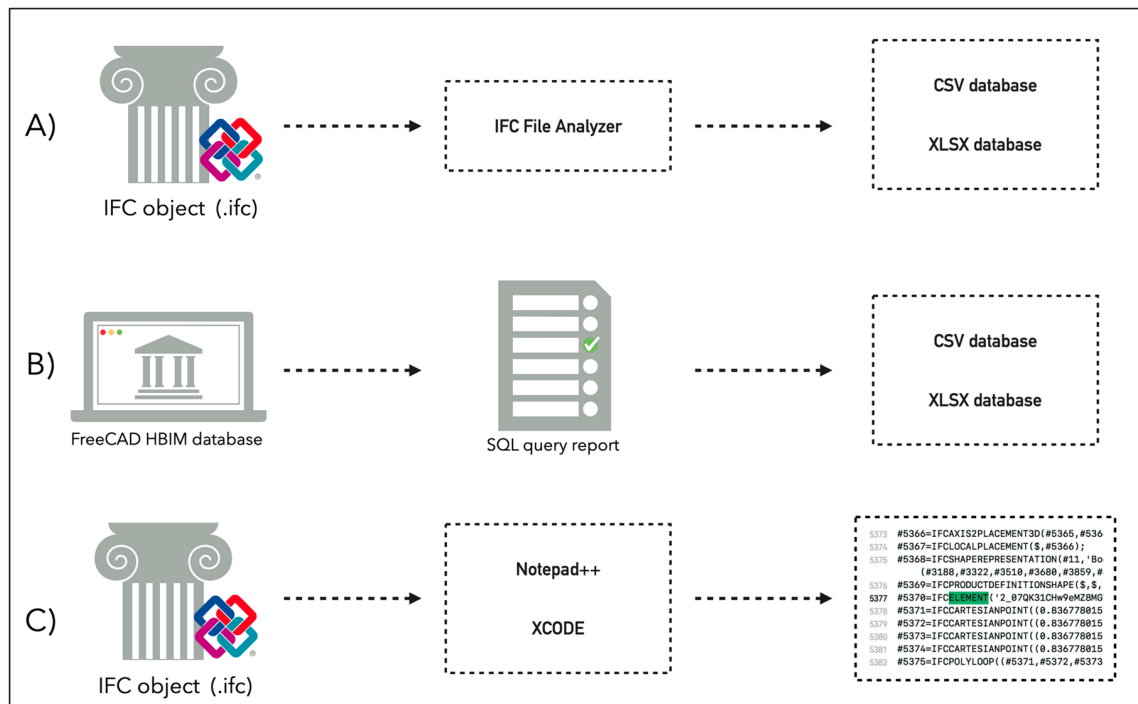

Figure 19. IFC objects external investigations summary: (**A**) IFC File Analyzer method; (**B**) SQL query report exportation; (**C**) IFC objects investigation and validation through code editors.

Another solution is based on DBMS. By exporting the datasheet output of SQL queries by using the CSV format, a complete report dataset can be obtained. Therefore, ad hoc queries and related results can spread externally the object knowledge, improving data exchange processes (Figure 20).

An additional solution to overcome interoperability and data exchange issues is based on the source code. Information concerning IFC BIM objects can be read and modify directly inside source codes by opening IFC files with code editors, such as Notepad++ for Windows and XCode for MacOS. Notepad++ is a full-featured text editor with features like syntax highlighting and syntax folding, user-defined syntax highlighting and folding, PCRE (Perl Compatible Regular Expression) search/replace, customizable GUI (graphic user interface) with vertical tab and vertical document list, document map, word completion, macro recording and playback, launch with different arguments, and more. XCode is the Apple-integrated development environment (IDE), including a text editor, a compiler and a build system into one software package.

By using code editor tools, the complete access and management of the IFC file become possible: properties such as description, roles and classifications can be easily read and modified. This approach is useful especially after the creation of models in order to set and fix some IFC parameters and to verify the whole source code of the IFC file object.

Concluding this section, the here presented IFC custom classification (small example) has been designed for particular heritage assets (mostly building archaeology), which hardly fits with other heritage peculiarities. However, we have seen how FreeCAD can adapt its libraries and tools to specific analyses: for example, other extended IFC classifications could be performed regarding different types of historical architectures as well as world heritage assets, allowing the management of new HBIM

analyses through IFC file exploitation. Then, FOSS software, libraries, query languages can be adapted to particular analyses, avoiding the opposite.

Moreover, in addition to the IFC interoperability (which remains fundamental), archaeological studies based on informative systems should continue to rely on DBMS and datasheet solutions due to historical data complexity and variety. FreeCAD, through SQL query possibilities, proved to be a dynamic informative platform for managing archaeological data in a single software.
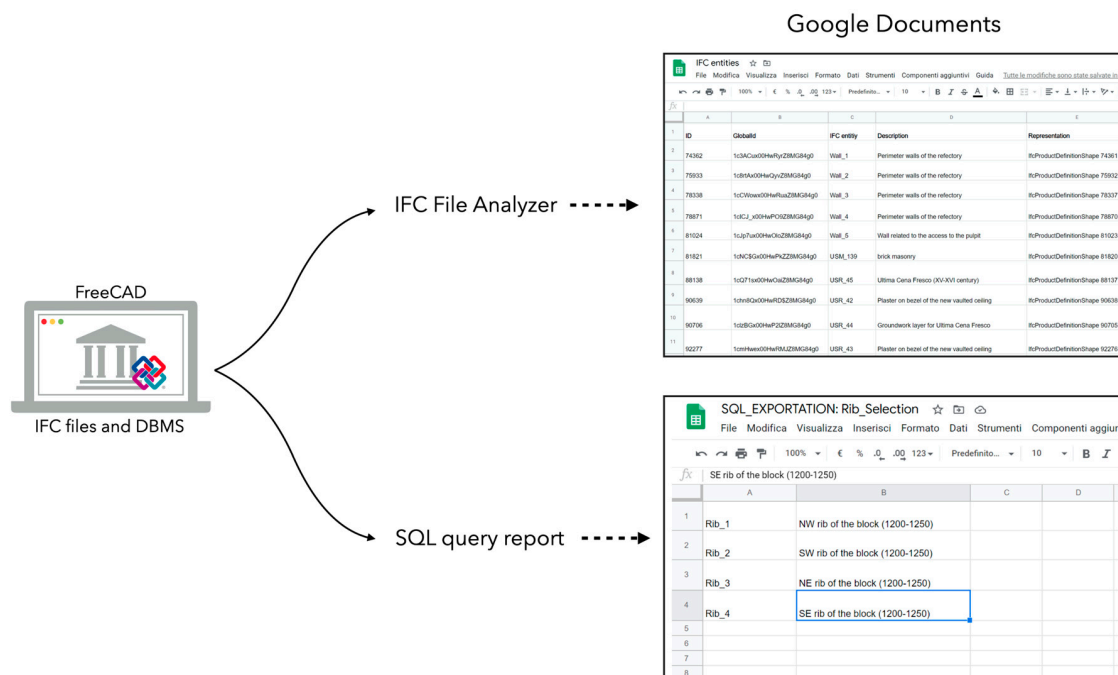

Figure 20. Example of exportable datasheets from IFC File Analyzer and SQL query report derived from IFC files and DBMS of FreeCAD software.

5. Results

The presented work is based on a problem-solving ad hoc process for cultural heritage assets IFC classification through the utilization of FreeCAD FOSS solution.

A demonstrative and basic classification has been designed for historic architecture as well as archaeological stratigraphy referring to medieval masonries, *Apsis—Arch—Architrave—Archivolt—Base—Brick—Buttress—Capital—Cladding Wall—Double-lancet Window—Pillar—Quoin—Retaining Wall—Rib—Roof Tile—Shaft—Shelf—Single-Lancet Window—Stone Block—Stone Element—Triple-Lancet Window—Vault*, have been included as far as historic architectural elements are concerned and *USM* (masonry stratigraphic unit), *USR* (render/plaster stratigraphic unit) and generic *EA* (architectural element) have been integrated for building archaeology purposes.

This operation has proven to be feasible by accessing the source code of Python files referring to FreeCAD ARCH workbench modules.

At the same time, query operations (through SQL language and SELECT syntax) have proven to work easily coupled with the proposed custom classification concerning particular examples (ionic column and capital, medieval rib vault, medieval stone wall). Then, the entire workflow designed for the purpose of managing ad hoc solutions for cultural heritage IFC assets correctly worked inside the FreeCAD environment (Figure 21), though it is not exportable into other BIM platforms due to standard limitations. However, this approach cannot be considered limited. Thanks to additional features of FreeCAD, a smart environment for historical data investigation can be created, including a customized IFC definition for a local BIM user experience.
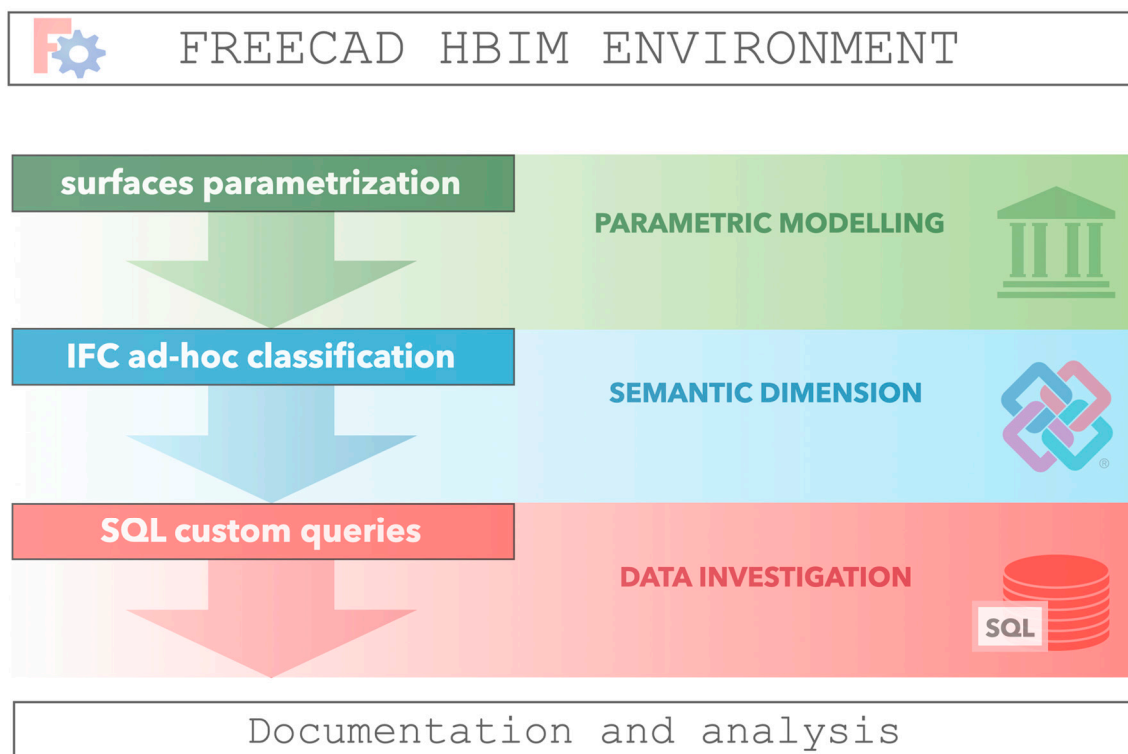
Figure 21. FreeCAD HBIM environment for documentation and analysis: from 3D parametric modeling based on predefined libraries and parametrization of imported surfaces; ad hoc semantic dimension through custom IFC classification and other attributes; ad hoc SQL queries for data investigation.

For this reason, the additional and customized *Reporting workbench* played a key role in validating ad hoc classification, because it allows for designing SQL queries and managing the obtained results in dynamic DBMS, easy to export (permitting data validation for collaborative approach). The correct semantic syntax of SQL queries has been verified, proving to be compatible with IFC schema modifications. Moreover, *Reporting workbench* can be exploited for the purpose of extracting any kind of information from object attributes, e.g., materials, descriptions, identification codes and *tags* (which can be used for extending textual information concerning building archaeological data). These queries and analyses become fundamental as far as the history of architecture is concerned—for example, for data verification, revisions and updates.

However, these unusual modifications (unplanned by standards of BuildingSMART) cannot be available externally (on web servers, viewers and management software). Web solutions and viewers are based on ISO standards (especially based on predefined queries on standard IFC families) [28], even if IFC objects can be managed by alternative solutions mostly referring to data extraction from objects and query output through DMBS.

In addition to limitations concerning IFC semantic information, other issues arose from this initial proposal. Despite custom roles relying on principal macro entities like ifcWall, ifcWindow, ifcRoof and so on, they need specific and accurate metric, physical and structural attributes and properties, because for this evaluative custom inclusion, they leaned on general attributes of their parent entities.

Moreover, the modeling procedure affecting BIM software involves an accurate analysis of custom components and classification. Being a method based on predefined parametric and theoretical libraries (e.g., walls, windows, roofs based on closed standards for AEC industry), it easily fits to default IFC schemas (IFC2x3 and IFC4.1), designed for the same goal. However, the default modeling technique as well as standard IFC classification are not the best solutions for cultural heritage assets. For this reason,

they require ad hoc modeling (free form modeling could be the proper way) and custom semantic classification to associate them.

One of the principal issues regards the creation of cultural heritage peculiarities starting from predefined parametric families, which is almost impossible due to unique characteristics: to overcome this issue, the ultra-simplified parametric modeling as well as the free form modeling (e.g., by using NURBS modeling) of particular features and the following parametrization of surfaces may represent a valid solution [13–29].

Being semantically linked, the predefined parametric modeling procedure inherits the IFC classification problem: predefined parametric families rely on standard IFC entities and vice versa. The complete lack of semantic IFC classification concerning cultural heritage assets is a pressing issue. This work attempted to address this delicate situation by proposing a demonstrative classification referring to particular peculiarities linked to historic architecture and archaeology, even if there are several problems still to be solved.

In conclusion, this project has been designed in order to provide a proper semantic classification for particular heritage asset peculiarities. In fact, the data analysis and validation of this project have been designed for archaeological studies. Archaeologists, by using FreeCAD software with additional features and custom modification, are able to reproduce parametrically the surveyed context; they are able to include archaeological and historical information; they are able to investigate digital models and related information through custom analyses and queries. Moreover, data exchange processes are ensured by exporting IFC models, datasheets and included semantic information.

Apart from GIS systems, archaeological research can scarcely rely on a single and smart platform for collecting, documenting and analyzing data, preventing data fragmentation. Customized FreeCAD software can be included inside the informative ecosystem considered for heritage assets.

6. Discussion

Beyond this custom and proposed classification (also intended as a theoretical suggestion for the future implementation of the BuildingSMART IFC schemas), the cultural heritage domain requires increasing attention concerning informative systems and, in turn, BIM possibilities and analyses.

Despite FreeCAD implementation via Python files related to ARCH modules, the integration and implementation of particular cultural heritage entities on IFC2X3 and IFC4.1 schema depends on the definition of specific international standards, including the adoption inside countries and groups of specialists.

These standardization procedures could be very long and time-consuming, even if from a parametric modeling point of view, it is impossible to achieve since all cultural heritage peculiarities do not have sharable design features.

In an attempt to solve the ongoing issues, a basic (and simplified) classification for cultural heritage entities could be examined by BuildingSMART, since the importance of BIM informative systems is not limited to the AEC industry. This methodology, being essentially a three-dimensional DBMS, includes many other applications useful for documentation and monitoring processes of heritage assets (e.g., historic architecture, archaeology, etc.).

7. Conclusions

The presented proposal, apart from unsolved issues, could be extremely useful for collecting semantic historical data inside the FreeCAD environment (Figure 21). This software offers the possibility to modify default libraries and schemas related to particular aspects of BIM objects.

By following FOSS modification and adaptation possibilities, the development of a custom-made workbench inside FreeCAD is a promising opportunity for historical buildings or archaeological sites. The previously mentioned roles/types could be included in a custom tool or additional workbench based on stratigraphic evidence (USM and USR) as well as for architectural elements (EA), as parametric surfaces linked to the IFC model, and their interpretation.

Consequently, following the proposed modifications and carrying out an HBIM project for building archaeology, the recording of the stratigraphic evolution could be possible also including specific description unit by unit.

The creation of an open database concerning definition of particular objects and related properties (structural as well as decorative) for the cultural heritage domain [32] represents a fundamental tool. BIM custom workflows and proposals could become more and more important for ad hoc documentation and monitoring processes, keeping safe and under control the knowledge sphere of cultural heritage assets. Despite the here presented demonstrative ad hoc implementation, this work has proven that FreeCAD software could be more flexible than other BIM closed software: in this way, different types of historical information can be therefore stored and managed in a single HBIM platform. Moreover, although this work should be intended as a proposal and theoretical extension of IFC schemas (also due to rigid AEC standardization), other building archaeology studies based on BIM methodology may rely on FreeCAD also regarding personal and proposal classification concerning objects classes, in order to achieve a fully operative HBIM environment.

References

1. De Luca, L.; Busayarat, C.; Stefani, C.; Véron, P.; Florenzano, M. A semantic based platform for the digital analysis of architectural heritage. *Comput. Graph.* **2011**, *35*, 227–241. [CrossRef]

2. Messaoudi, T.; Véron, P.; Halin, G.; De Luca, L. An ontological model for the reality-based 3D annotation of heritage building conservation state. *J. Cult. Herit.* **2018**, *29*, 100–112. [CrossRef]

3. Garba, S.B.; Hassanain, H.M. A review of object oriented cad potential for building information modelling and life cycle management. In Proceedings of the 1st ASCAAD International Conference, e-Design in Architecture, KFUPM, Dhahran, Saudi Arabia, 7–9 December 2004; pp. 343–359.

4. Tse, T.K.; Wong, K.A.; Wong, K.F. The utilisation of building information models in nD modelling: A study of data interfacing and adoption barriers. *J. Inf. Technol. Constr.* **2005**, *10*, 85–110.

5. National Building Information Standards. Available online: https://www.nationalbimstandard.org/ (accessed on 20 November 2020).

6. Diara, F.; Rinaudo, F. Open source HBIM for Cultural Heritage: A project proposal. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Copernicus GmbH: Göttingen, Germany, 2018; Volume XLII-2, pp. 303–309.

7. BuildingSmart. Available online: https://www.buildingsmart.org/ (accessed on 20 November 2020).

8. B-Cert, the Official Platform for the BuildingSmart IFC4 Certification. Available online: https://www.b-cert.org/ (accessed on 20 November 2020).

9. ISO 16739-1. Industry Foundation Classes (IFC) for Data Sharing in the Construction and Facility Management Industries—Part 1: Data Schema, ISO 16739-1:2018. Available online: https://www.iso.org/standard/70303.html (accessed on 20 November 2020).

10. ISO/TC 184/SC 4. Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 21: Implementation Methods: Clear Text Encoding of the Exchange Structure, ISO 10303-21:2016. Available online: https://www.iso.org/standard/63141.html (accessed on 20 November 2020).

11. Lee, G.; Jeong, J.; Won, J.; Cho, C.; You, S.J.; Ham, S.; Kang, H. Query performance of the IFC model server using an object-relational database approach and a traditional relational database approach. *J. Comput. Civ. Eng.* **2014**, *28*, 210–222. [CrossRef]

12. BuildingSmart IFC Specifications. Available online: https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/ (accessed on 20 November 2020).

13. Diara, F.; Rinaudo, F. From reality to parametric models of Cultural Heritage assets for HBIM. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Copernicus GmbH: Göttingen, Germany, 2019; Volume XLII-2, pp. 413–419.

14. Murphy, M.; McGovern, E.; Pavia, S. Historic building information modelling (HBIM). *Struct. Surv.* **2009**, *27*, 311–327. [CrossRef]

15. Murphy, M.; McGovern, E.; Pavia, S. Historic Building Information Modelling—Adding intelligence to laser and image based surveys of European classical architecture. *ISPRS J. Photogramm. Remote Sens.* **2013**, *76*, 89–102. [CrossRef]

16. Fassi, F.; Parri, S. Web 3D BIM for the Department of Cultural Heritage. Tests and experimentation on the Spire of the Duomo of Milan. In *To Know, Conserve, Enhance (a Cura di ROSA Anna Genovese)*; Genovese, R.A., Ed.; Arte Tipografica Editrice: Napoli, Italy, 2013; pp. 271–290.

17. Adami, A.; Scala, B.; Spezzoni, A. Modelling and accuracy in a BIM environment for planned conservation: The apartment of Troia of Giulio Romano. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Copernicus GmbH: Göttingen, Germany, 2017; Volume XLII-2/W3, pp. 17–23.

18. Logothetis, S.; Stylianidis, E. BIM Open Source Software (OSS) for the documentation of Cultural Heritage. *Virtual Archaeol. Rev.* **2016**, *7*, 28–35. [CrossRef]

19. Logothetis, S.; Karachaliou, E.; Stylianidis, E. From OSS CAD to BIM for Cultural Heritage digital representation. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Copernicus GmbH: Göttingen, Germany, 2017; Volume XLII-2/W3, pp. 439–445.

20. Logothetis, S.; Karachaliou, E.; Valari, E.; Stylianidis, E. Open source cloud-based technologies for BIM. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *XLII-2*, 607–614. [CrossRef]

21. Simeone, D.; Cursi, S.; Toldo, I.; Carrara, G. BIM and knowledge management for building heritage. In Proceedings of the ACADIA 2014 Design Agency, Los Angeles, CA, USA, 23–25 October 2014; pp. 681–690.

22. Scianna, A.; Serlorenzi, M.; Gristina, S.; Filippi, M.; Paliaga, S. Sperimentazione di tecniche BIM sull'archeologia romana: Il caso delle strutture rinvenute all'interno della cripta della chiesa dei SS. Sergio e Bacco in Roma. *Archeol. Calcolatori Suppl.* **2015**, *7*, 199–212.

23. The FreeCAD Team, FreeCAD. Available online: https://www.freecadweb.org/ (accessed on 20 November 2020).

24. OpenShell Library. Available online: http://ifcopenshell.org/python.html (accessed on 20 November 2020).

25. Brogiolo, G.P. *Archeologia Dell'edilizia Storica: Documenti e Metodi*; Brogiolo, G.P., Zigrino, L., Zonca, A., Eds.; New Press: Como, Italy, 2013.

26. Li, H.; Liu, H.; Liu, Y.; Wang, Y. An object-relational IFC storage model based on oracle database. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *XLI-B2*, 625–631. [CrossRef]

27. Quattrini, R.; Battini, C.; Mammoli, R. HBIM to VR. Semantic awareness and data enrichment interoperability for parametric libraries of historical architecture. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Copernicus GmbH: Göttingen, Germany, 2018; Volume XLII-2, pp. 937–943.

28. Tauscher, E.; Bargstädt, H.J.; Smarsly, K. Generic BIM queries based on the IFC object model using graph theory. In Proceedings of the 16th International Conference on Computing in Civil and Building Engineering, Osaka, Japan, 6–8 July 2016.

29. Oreni, D.; Brumana, R.; Banfi, F.; Bertola, L.; Barazzetti, L.; Cuca, B.; Previtali, M.; Roncoroni, F. Beyond Crude 3D Models: From Point Clouds to Historical Building Information Modeling via NURBS. *Lect. Notes Comput. Sci.* **2014**, *8740*, 166–175.

30. Lipman, R.R. Developing Coverage Analysis for IFC Files. In Proceedings of the CIB W78 27th International Conference, Cairo, Egypt, 16–18 November 2010.

31. Logothetis, S.; Valari, E.; Karachaliou, E.; Stylianidis, E. *Spatial DMBS architecture for a free and open source BIM, In The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Proceedings of the 26th International CIPA Symposium 2017, Ottawa, ON, Canada, 28 August–1 September 2017*; Copernicus GmbH: Göttingen, Germany, 2017; Volume XLII-2/W5, pp. 467–473. [CrossRef]

32. Oreni, D.; Brumana, R.; Georgopoulos, A.; Cuca, B. HBIM Library Objects for Conservation and Management of Built Heritage. *Int. J. Herit. Digit. Era* **2014**, *3*, 321–334. [CrossRef]