Implementing Information Delivery Specification (IDS) encoding for a BIM object standard based on model uses

Sthefani Bigai, sthefanibigai@gmail.com
Universidade de São Paulo (USP), São Paulo, Brazil

Eduardo Toledo Santos, etoledo@usp.br
Universidade de São Paulo (USP), São Paulo, Brazil

Abstract
The lack of clearly defined and validated information requirements in model-based projects significantly limits the benefits of Building Information Modelling (BIM). This study addresses this issue by examining the Information Delivery Specification (IDS), a new open BIM standard developed by buildingSMART for specifying exchange information requirements in BIM models. Drawing upon the Brazilian Standard for BIM Objects, which is still under development and aims to establish requirements for BIM object creation based on various model uses, this paper presents an approach to using the IDS standard to encode attribute requirements related to model uses. The research employs the Brazilian Association of Technical Standards' (ABNT) Standard for BIM Objects as a foundation and demonstrates how it is possible to use IDS to bridge the gap between the model use-based requirements and the IFC schema, that is based on object types. The results indicate that employing open BIM standards like IDS enhances information management efficiency throughout BIM project workflows, promoting more reliable data exchange and coordination. This study contributes to the ongoing efforts to optimize BIM implementation by providing a structured and automated approach to defining and validating information requirements.

Keywords: IDS, BIM, BIM Object

1 INTRODUCTION

Building Information Modelling (BIM) refers to a business process for generating and leveraging building data to design, construct and operate the building during its lifecycle. BIM allows all stakeholders to have access to the same information at the same time through interoperability between technology platforms (NBIMS 2007).

Despite the increasing worldwide adoption of BIM, the lack of definition of information requirements and, later, their validation in model-based project deliveries remains a significant barrier to both owners and designers to achieving all potential benefits offered by BIM. The literature identifies two main challenges that contribute to this barrier: the lack of resources and knowledge to assess the quality of BIM models, and the owners' insufficient understanding of the information needed to support the asset's lifecycle (Cavka et al 2017).

Aiming to improve the predictability and reliability of data exchange workflows, buildingSMART created the IDS (Information Delivery Specification) standard. It is a computer interpretable document standard designed to specify the Exchange Information Requirements or (partially) the Level of Information Need for IFC (Industry Foundation Classes) models. Using IDS allows for specifying how properties, classifications, and values should be entered and exported in a BIM model, enabling the automated validation of the project's required information (Berlo et al 2023).

The Brazilian Association of Technical Standards (ABNT) is currently authoring a new technical standard for BIM objects to foster the creation of object libraries of construction components available in the internal market. Considering previous international initiatives and standards with similar goals - e.g. SPie (Kalin & Weygant 2013), BIM Hawk (NG Bailey 2016), PDTs, etc. - which failed or are slowly advancing due to the huge quantity of different types of components, this ABNT standard intends to establish requirements for objects using a novel approach: instead of categorizing requirements based on the object type (e.g. door, pipe, light fixture, etc.) it specifies the minimum object requirements according to the intended model use(s) (e.g., clash detection, quantity take off, energy analysis, 4D planning, etc.) for the BIM model where those objects are instantiated (Santos & Catelani 2017). This development approach requires recruiting only a few model use experts (currently, 18 model uses are selected for the standard) instead of a single large group working out the product data information requirements for thousands of object types and subtypes. It is up to the object developers using the ABNT standard to decide which model uses are relevant for a type of component and their object library users. Although not devised for this purpose, this standard is likely to be used by owners as a guide for specifying the information requirements linked to each model use that they intend to adopt for their projects, alleviating the aforementioned problem.

Like IDS, the ABNT Standard for BIM Objects is IFC-based (i.e., information requirements are specified in terms of IFC entity attributes and properties) hence, it is possible to express the requirements in the standard using IDS.

However, the IFC schema is heavily based on object types, but object types are not directly mentioned in the ABNT standard (since it is object type neutral and based on model usage). Instead, linked to each required information attribute in the standard, there are "if-clauses"/conditions on component characteristics to determine if the attribute is mandatory or not, in each case. For example, for the model use "energy analysis", the attribute "*power*" is required "*if the component consumes energy*". Therefore, the *power* attribute is mandatory for a heater or a light source, but not for a table or a wall object.

Accordingly, this paper introduces an approach to utilize the IDS standard to encode attribute requirements related to model uses, based on the ABNT Standard for BIM Objects, overcoming the conceptual difference between the two standards regarding object typing. Having .ids files for each of the parts of the ABNT standard allows easy validation of information requirements in BIM models, according to the selected model uses for the project.

2 RESEARCH METHODOLOGY

The methodology used in this paper is based on the Design Science Research approach, a problem-solving-oriented method focused on creating and evaluating artifacts designed to transform situations (Dresch et al 2015).

The research was conducted in three main stages. Having defined the problem to solve (i.e., mitigate the information requirements specification gap) the first stage involved a literature review covering key topics related to defining information requirements in BIM projects. Considering the opportunity to investigate the use of an emerging new open BIM standard, a more detailed review was conducted on Information Delivery Specification (IDS), which is central to this work.

The second stage focused on creating the artifact: .ids files encoding the ABNT Standard for BIM Objects requirement for selected model uses. For this paper, a single IDS file was developed - architectural design modelling - one of the BIM uses outlined in the ABNT standard. For the development of the IDS file, the IDS Editor tool, part of the ACCA Software's US BIM platform (ACCA Software [n.d.]), was adopted to facilitate the creation of the IDS file. This platform was chosen for its ability to enable the creation of XML files in a clear and organized manner, thereby minimizing errors and simplifying comprehension during the development process. Additionally, the platform is freely accessible, allowing users to download it at no additional cost.

The third stage is the artifact's validation. The IDS file was tested to demonstrate its functionality, using a quality checklist for a specific BIM model, based on the requirements of the ABNT standard. For the validation of the IDS file, the BlenderBIM platform (BlenderBIM [n.d]) was used, which allows the verification of IFC model requirements through the use of IDS files. Similar to the US BIM platform, BlenderBIM is free and open source.

3 LITERATURE REVIEW

3.1 IDS (Information Delivery Specification)

The Information Delivery Specification (IDS) is a new open BIM standard initially proposed in 2020 by buildingSMART, with its first version released in June 2024. It is an XML-based standard that can be interpreted by computers, designed to code information exchange requirements based on the IFC schema (Berlo et al 2023).

IDS is an open BIM standard, which means it supports digital workflows based on vendor-neutral formats such as IFC, BCF, and COBie. This way, IDS was developed to promote structured and standardized management of information requirements while enabling optimized validation of IFC models.

An IDS file contains a set of specifications, each one representing a specific information requirement. Each specification is divided into three sections, represented in the file by XML tags: Description, Applicability, and Requirements. The Description section is intended for project stakeholders, explaining what information is being requested. The Applicability and Requirements sections focus on computer interpretation, structured with XML tags. The Applicability section identifies the IFC entities to which the specification applies, while the Requirements section outlines the specific information needed for the entities identified in the previous section.

The Applicability and Requirements sections are described using facets, which consist of predefined XML tags. Through these tags, the computer can interpret the specified data in a precise and structured manner, enabling the description of the information that an entity of the model must contain. In addition to requirements, IDS verifies the formatting, naming, and encoding of model data, enabling designers to generate models compliant with the specified requirements.

Figure 1 shows an example of specification using IDS.

```xml
<ids:specification ifcVersion="IFC2X3 IFC4" name="Spec 1" minOccurs="1" maxOccurs="1"
    description="the model must have rooms, every room must have a name from a list of allowed values (RN1, RN2...)">
    <ids:applicability>
        <ids:entity>
            <ids:name>
                <ids:simpleValue>IFCSPACE</ids:simpleValue>
            </ids:name>
        </ids:entity>
        <ids:classification>
            <ids:value>
                <ids:simpleValue>Room</ids:simpleValue>
            </ids:value>
        </ids:classification>
    </ids:applicability>
    <ids:requirements>
        <ids:attribute minOccurs="1" maxOccurs="1">
            <ids:name>
                <ids:simpleValue>Name</ids:simpleValue>
            </ids:name>
            <ids:value>
                <xs:restriction base="xs:string">
                    <xs:pattern value="RN." />
                </xs:restriction>
            </ids:value>
        </ids:attribute>
    </ids:requirements>
</ids:specification>
```

Figure 1. IDS file example.

The current (first) version of IDS does not support geometric checks, validations based on dynamic values, or assessments of information not covered by the IFC schema. Nevertheless, IDS can specify and validate key information about model elements, as well as relationships between entities, in accordance with the IFC schema.

The process for using IDS in creating and validating information requirements consists of three main stages. The first stage involves the creation of the IDS by the client, according to the project's information requirements. In the second stage, the IDS file is used by the designer as a guide to create the data, validate the model, and export the IFC file according to the client's specifications. The third stage involves the client validating the received model using the IDS file, ensuring that the established requirements are met.

4 IDS DEVELOPMENT

The creation of the IDS file involved a two-step process based on its sections: first, defining the Requirements section, then, defining the Applicability section for each parameter.

For this paper, the IDS file was developed based on the part 2 of the ABNT standard, which has already been finalized. The Part 2 of the standard defines parameters for the model use "Architecture Design", along with the criteria for their compulsory inclusion in the object and parameter type (instance or type). Eight essential parameters for BIM objects are specified, including object dimensions, structural characteristics, and its main material. The parameters are detailed in Table 1. The part 1 of the standard specifies 8 additional parameters, common to all model uses, which will not be covered in this paper.

Table 1. Parameters defined by the Part 2 of the ABNT standard and inclusion criteria.

Part 2 – Architectural Design		
Parameter	**Description**	**Criteria**
Component Description	Generic description of the component - may include information on type, physical characteristics, brand, dimensions, and other features that define the component.	Mandatory for all component types, except terrain / site.
Component ID	Name of the component in the project context.	Mandatory for all designed components.
Internal/External	Indicates whether the component is located inside or outside the building.	Mandatory for components that may be affect by the weather.
Structural/ Non-Structural	Indicates whether the component is structural (S) or non-structural (N).	Mandatory for structural components.
Height	Vertical dimension of the component or its thickness when applicable.	Mandatory.
Width	Transverse dimension of the component.	Mandatory.
Length	Longitudinal dimension of the component.	Mandatory.
Material	Main material of the component.	Mandatory.

4.1 Requirements Part

For every parameter outlined within the ABNT standard, it is necessary to identify the associated attribute or property within the IFC schema that corresponds to it. This information is already included in the text of the standard, as outlined in Table 2.

Table 2. IFC Attributes/Properties mapped for each parameter.

Parameter	**IFC Attribute/Property**
Component Description	Description
Component ID	Name
Internal/ External	IsExternal (Property set: Common)
Structural/ Non-Structural	LoadBearing (Property set: Common)
Height	Height or Depth (Property set: BaseQuantities)
Width	Width (Property set: BaseQuantities)
Length	Length (Property set: BaseQuantities)
Material	N/A (native relationship of IFC)

After defining the IFC attributes or properties, the process of "translating" to the IDS standard began, employing the predefined facets. The attributes *Name* and *Description* were articulated using the <attribute> facet, designed to delineate the fundamental data associated with all IFC entities, as depicted in in Figure 2.

```
<ids:requirements>
  <ids:attribute>
    <ids:name>
      <ids:simpleValue>Name</ids:simpleValue>
    </ids:name>
  </ids:attribute>
</ids:requirements>
```

```
<ids:requirements>
  <ids:attribute>
    <ids:name>
      <ids:simpleValue>Description</ids:simpleValue>
    </ids:name>
  </ids:attribute>
</ids:requirements>
```

Figure 2. Requirements Section for the attributes Name and Description.

In a readable format, this specification indicates that "The entities mentioned in the Applicability section must meet the requirement of *having the Name/Description attribute*".

The properties *IsExternal* and *LoadBearing* were defined using the facet <property>, as they are data attached to the IFC objects and organized into Property Sets. These properties are part of a standardized Property Set, denoted by the prefix "Pset_" and suffixed with "Common". Given that the name of the property set varies depending on the entity, its designation was specified using complex constraints. Consequently, a pattern restriction was employed utilizing the symbol ".*", which serves as a wildcard representing any number of characters. It is essential to establish the data type of the value for properties as well. In this case, the data type selected for both properties were IfcBoolean. The requirements section for these properties is shown in Figure 3.

```
<ids:requirements>
  <ids:property measure="IfcBoolean" minOccurs="1" maxOccurs="unbounded">
    <ids:propertySet>
      <xs:restriction base="xs:string">
        <xs:pattern value=".*Common" />
      </xs:restriction>
    </ids:propertySet>
    <ids:name>
      <ids:simpleValue>IsExternal</ids:simpleValue>
    </ids:name>
  </ids:property>
</ids:requirements>
```

```
<ids:requirements>
  <ids:property measure="IfcBoolean" minOccurs="1" maxOccurs="unbounded">
    <ids:propertySet>
      <xs:restriction base="xs:string">
        <xs:pattern value=".*Common" />
      </xs:restriction>
    </ids:propertySet>
    <ids:name>
      <ids:simpleValue>LoadBearing</ids:simpleValue>
    </ids:name>
  </ids:property>
</ids:requirements>
```

Figure 3. Requirements Section for the properties IsExternal and LoadBearing.

The properties *Height*, *Depth*, *Width*, and *Length* were treated using the same logical approach as described previously for the <property> facet. The main difference lies in their association with a Quantity Set, indicated by the prefix "Qto_" and suffixed with "BaseQuantities". Additionally, the datatype employed for these properties is IfcLengthMeasure, as shownin Figure 4.

```
<ids:requirements>
  <ids:property measure="IfcLengthMeasure" minOccurs="1" maxOccurs="unbounded">
    <ids:propertySet>
      <xs:restriction base="xs:string">
        <xs:pattern value=".*BaseQuantities" />
      </xs:restriction>
    </ids:propertySet>
    <ids:name>
      <ids:simpleValue>Height</ids:simpleValue>
    </ids:name>
  </ids:property>
</ids:requirements>
```

Figure 4. Requirements Section for the property Height.

Finally, the *Material* property is an inherent aspect of the IFC schema, characterized by a specific facet known as <material>, as illustrated in Figure 5. Leveraging this facet enables the verification of whether entities possess a defined material attribute.

```
<ids:requirements>
  <ids:material minOccurs="1" maxOccurs="unbounded" />
</ids:requirements>
```

Figure 5. Requirements Section for the Material.

As a result, the Requirements section of the IDS file has been completed.

4.2 Applicability Part

The Applicability section represents the main challenge in the research reported in this paper, as there is the need to identify the object types that must be checked for requirements but, as mentioned earlier, they are not specified in the ABNT standard. Therefore, a mapping of the inclusion criteria to IFC entities must be developed, according to their usual characteristics.

The IFC (Industry Foundation Classes) schema is structured around three fundamental concepts represented by three classes: IfcObjectDefinition, IfcPropertyDefinition and IfcRelationship.

In this paper, the focus was on the IfcObjectDefinition concept, which is responsible for representing the objects needed to describe a construction asset. However, the term "object" can encompass anything from physically tangible elements to processes. To focus exclusively on physical elements, the emphasis was placed on the entities within the hierarchy under the IfcElement class, as depicted in Figure 6, representing physically existing elements within the construction.

Figure 6. IfcElement hierarchy (adapted from buildingSMART).

Entities below the IfcElement class in the hierarchy were selectively excluded from consideration. This included classes such as IfcCivilElement, IfcElementComponent, IfcFeatureElement, and IfcVirtualElement. They were disregarded either due to their deprecated status (IfcCivilElement) or because they represent items or elements that hold minimal relevance for architectural design.

Apart from the entities representing object instances (those included in the context of geometric representation in the project), entities representing object types are also relevant to this study. Object types define the specific information common to all instances of elements of the same type. Accordingly, elements derived from the IfcElementType class are also pertinent for the development of the proposed IDS file.

With this initial filter in place, an analysis was conducted for each parameter to determine which entities possess the attributes or properties outlined in the Requirements section and fall under the criteria defined in the ABNT standard. The parameters Description and Name are attributes of the IfcRoot entity and, therefore, belong to all subclasses. The IfcRelAssociatesMaterial relationship, that is used to associate materials to objects, can be applied to the IfcElement and IfcElementType classes. For the remaining parameters, an analysis of the IFC scheme was undertaken to identify the entities that include the corresponding properties and falls under the criteria defined in the ABNT standard. The distinction between the type (parameter holds a common value for all instances) or instance (parameter value can be different in each instance) nature of each parameter is also defined in the ABNT standard and shown in Table 3. Finally, a specification was formulated for each parameter, detailing the relevant entities.

Table 3. IFC Classes mapped for each parameter.

Parameter	Type (T)/ Instance (I)	IFC Classes
Component Description	T	Subclasses of: IfcBuiltElementType, IfcDistributionElementType, IfcElementAssemblyType, IfcFurnishingElementType, IfcGeographicElementType, IfcGeotechnicalElementType, IfcTransportationDeviceType
Component ID	I	Subclasses of: IfcBuiltElement, IfcDistributionElement, IfcElementAssembly, IfcFurnishingElement, IfcGeographicElement, IfcGeotechnicalElement, IfcTransportationDevice
Internal/ External	I	IfcRamp, IfcDoor, IfcColumn, IfcChimney, IfcCurtainWall, IfcCovering, IfcShadingDevice, IfcBeam, IfcJunctionBox, IfcMember, IfcStair, IfcPlate, IfcSlab, IfcWall, IfcRoof, IfcRailing, IfcBuildingElementProx, IfcWindow
Structural/ Non-Structural	I	IfcRamp, IfcColumn, IfcPile, IfcChimney, IfcBeam, IfcMember, IfcStair, IfcPlate, IfcFooting, IfcSlab, IfcWall, IfcBuildingElementProxy
Height	I	IfcCurtainWall, IfcDoor, IfcFooting, IfcKerb, IfcWall, IfcWindow, IfcPavement, IfcVehicle, IfcJunctionBox, IfcDistributionChamberElement, IfcSlab
Width	I	IfcCovering, IfcCurtainWall, IfcDoor, IfcFooting, IfcKerb, IfcPavement, IfcPlate, IfcRampFlight, IfcSlab, IfcWall, IfcWindow, IfcVehicle, IfcJunctionBox
Length	I	IfcBeam, IfcChimney, IfcColumn, IfcCurtainWall, IfcFooting, IfcKerb, IfcMember, IfcPavement, IfcRail, IfcRailing, IfcRampFlight, IfcSlab, IfcStairFlight, IfcWall, IfcVehicle, IfcDuctFitting, IfcJunctionBox, IfcPipeFitting, IfcCableCarrierSegment, IfcCableSegment, IfcDuctSegment, IfcPipeSegment, IfcSpaceHeater
Material	T	Subclasses of: IfcBuiltElementType, IfcDistributionElementType, IfcElementAssemblyType, IfcFurnishingElementType, IfcGeographicElementType, IfcGeotechnicalElementType, IfcTransportationDeviceType

For representation in the IDS file, the IFC classes were enumerated using the <entity> facet within the Applicability section, as illustrated in Figure 7.

```
<ids:applicability>
  <ids:entity>
    <ids:name>
      <xs:restriction base="xs:string">
        <xs:enumeration value="IFCBEAMTYPE" />
        <xs:enumeration value="IFCBEARINGTYPE" />
        <xs:enumeration value="IFCBUILDINGELEMENTPROXYTYPE" />
        <xs:enumeration value="IFCCHIMNEYTYPE" />
        <xs:enumeration value="IFCCOLUMNTYPE" />
        <xs:enumeration value="IFCCOVERINGTYPE" />
        <xs:enumeration value="IFCCURTAINWALLTYPE" />
        <xs:enumeration value="IFCDEEPFOUNDATIONTYPE" />
```

Figure 7. Part of an Applicability section.

4.3 Description Part

To finalize the IDS file, the Description section was completed. For each parameter, a description of the applicable IFC version and instructions on how the parameter should be filled were included. Figure 8 shows an example of a description for the parameter *Name*.

```
<ids:specification ifcVersion="IFC2X3 IFC4" name="Name" minOccurs="0" maxOccurs="unbounded"
description="Parameter Name." instructions="Name of the object within the project.">
```

Figure 8. Description section for the parameter Name.

5 IDS VALIDATION

The validation of the IDS file was conducted by modeling a simple architectural project to test whether the IDS file properly verifies the ABNT standard's requirements. A simple house was modeled in Revit, as illustrated in Figure 9, consisting of five different object types and nine instances: four walls, one floor, one roof, two windows, and one door.

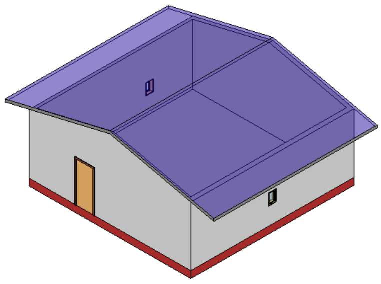

Figure 9. Architectural model in Revit.

Each parameter specified in the normative was checked using the IDS file developed against the corresponding entities, as outlined in Table 4.

Table 4. IFC Classes of the architectural model.

Parameter	IFC Classes	Total
Component Description	IfcWallType, IfcSlabType, IfcRoofType, Ifc WindowType, IfcDoorType	5
Component ID	IfcWall (x4), IfcSlab (x1), IfcRoof (x1), IfcWindow (x2), IfcDoor (x1)	9
Internal/ External	IfcWall (x4), IfcSlab (x1), IfcRoof (x1), IfcWindow (x2), IfcDoor (x1)	9
Structural/ Non-Structural	IfcWall (x4), IfcSlab (x1)	5
Height / Depth	IfcWall (x4), IfcWindow (x2), IfcDoor (x1) / IfcSlab (x1)	8
Width	IfcWall (x4), IfcSlab (x1), IfcWindow (x2), IfcDoor (x1)	8
Length	IfcWall (x4), IfcSlab (x1)	5
Material	IfcWallType, IfcSlabType, IfcRoofType, Ifc WindowType, IfcDoorType	5

For the initial verification, all parameters were defined within Revit, and thus, all entities were expected to pass the validation test. However, not all standard IFC object attributes have corresponding mappings to internal Revit data, and therefore, some attributes, such as *Description*, are not supported. To validate the IDS file, the *Description* attribute was manually added to the object types by editing the IFC file after exporting it.

The validation was performed using the BlenderBIM software. Figure 10 illustrates the results of the first test taken.

Figure 10. 1st test results by BlenderBIM.

Based on the initial test results, the IDS file accurately identified the IFC entities to be checked for each parameter. Furthermore, the verification of the parameters aligned with the expected results.

Subsequently, a second test was conducted to validate whether the IDS file correctly identifies when an attribute or property does not meet the requirements. For this test, the attributes and properties of the IfcSlab and IfcSlabType entities were removed, and the test was performed one more time using BlenderBIM. Figure 11 shows the results of the second test taken.

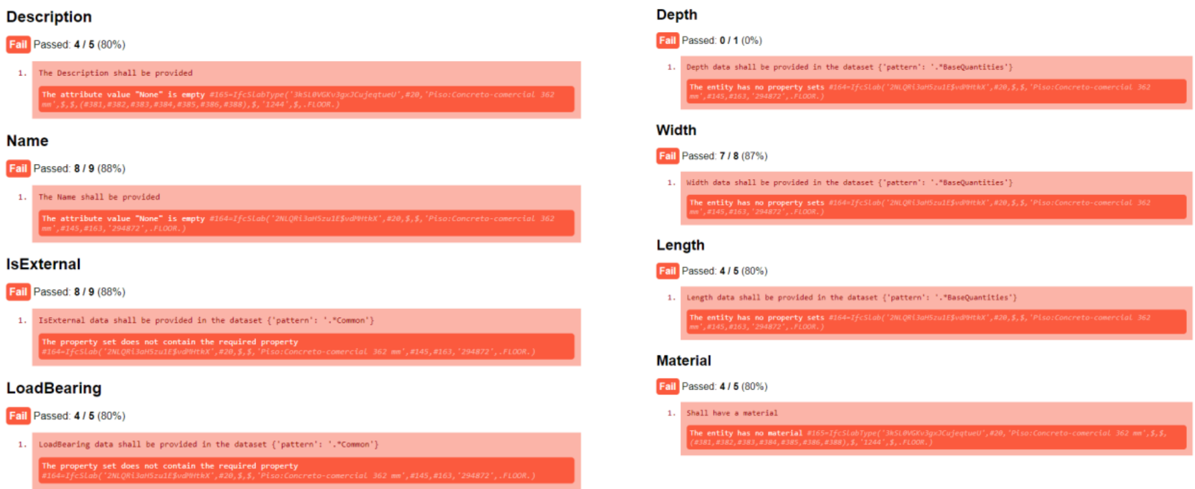

Figure 11. 2nd test results by BlenderBIM.

The results indicate that the IDS successfully verified the absence of the specified attributes and properties.

6 DISCUSSION

This research serves as a proof of concept for using open standards to validate information requirements in BIM projects. The developed artifact facilitates the verification and validation of the requirements established by the ABNT standard, utilizing open standards. This promotes more effective information management throughout the procurement and coordination workflow of BIM projects.

As the IDS standard is still under development, there are challenges related to platform compatibility with the newest versions: it is common to encounter incompatibilities between platforms used for creating IDS and those used for validation. However, with the release of the official version and the integration of IDS into the project review workflow, these issues are expected to be resolved.

The next step of the research will be: to extend the application of the developed artifact to encompass not only the Architecture Design model use, but also some of the other 17 uses outlined by the ABNT Object Standard. This process involves studying the requirements established for each use, mapping the properties within the IFC schema, and understanding the methodology for specification according to the IDS standard. Thus, more challenging criteria in the ABNT standard will require further analysis for implementation in the applicability section.

Furthermore, it is crucial to investigate emerging advancements in IDS. This in-depth study will enable the incorporation of new functionalities into the artifact, enhancing its applicability in the field of BIM project coordination.

7 CONCLUSIONS

The successful implementation of BIM is directly linked to the proper selection and execution of its uses. Therefore, it is recognized that both the development of an appropriate specification and effective validation are essential to ensure the fulfillment of established requirements.

Based on the studies and analyses conducted, it is evident that investigating the application of IDS in the procurement and coordination workflow of BIM models represents a valuable contribution to improving the efficiency of BIM use in the construction industry.

This research addresses the impact of defining and validating information requirements through structured and automated methods. The expected results have the potential to enhance the effectiveness of information exchanges in projects, thereby maximizing the benefits of BIM adoption.

References

ACCA Software. (n.d.). *ACCA BIM Software*. Available at: https://www.accasoftware.com/en/bim-software. Accessed on May 31, 2024.

Berlo, L. v., Drogemuller, R., & Omrani, S. (2023). *What is Information Delivery Specification (IDS)*. Available at: https://www.buildingsmart.org/what-is-information-delivery-specification-ids/. Accessed on May 31, 2024.

BlenderBIM Add-on. (n.d.). *BlenderBIM*. Available at: https://blenderbim.org/. Accessed on May 31, 2024.

Cavka, H. B., Staub-French, S., & Poirer, E. A. (2017). Developing owner information requirements for BIM-enabled project delivery and asset management. *Automation in Construction*, 83, pp. 169-183. Available at: https://www.elsevier.com/ locate/aei.

Dresch, A., Lacerda, D. P., & Valle Antunes Jr, J. A. (2015). *Design Science Research: A Method for Science and Technology Advancement*. Available at: https://link.springer.com/book/10.1007/978-3-319-07374-3.

Kalin, M. & Weygant, R. (2013). *Specifiers Properties information exchange (SPie) - Minimum Building Information Model (BIM) Object Definitions*. ERDC/CERL CR-13-1. US Army Corps of Engineers. DOI 10.21236/ada587606.

NBIMS (National Building Information Model Standard) Project Committee. (2007). *United States national building information modeling standard - Version 1 - Part 1: Overview, principles, and methodologies*. Washington, DC: National Institute of Building Sciences.

NG Bailey. (2016). *The BIMHawk Toolkit*. Available at: https://www.bimhawk.co.uk/. Accessed on May 31, 2024.

Santos, E. T. & Catelani, W. S. (2017). Normas brasileiras sobre BIM - parte 2: componentes BIM. *Concreto & Construções*, 85 (1), pp. 55-59. Available at: https://ibracon.org.br/Site_revista/Concreto_Construcoes/ebook/edicao85/files/assets/basic-html/index.html#55.