World Building Congress 2022 IOP Publishing
IOP Conf. Series: Earth and Environmental Science 1101 (2022) 092024 doi:10.1088/1755-1315/1101/9/092024

A review of methods to specify information requirements in digital construction projects

A Tomczak[1], **L v Berlo**[2], **T Krijnen**[3], **A Borrmann**[4] **and M Bolpagni**[5]

[1] Norwegian University of Science and Technology, Norway
[2] buildingSMART International, United Kingdom
[3] Delft University of Technology, Netherlands
[4] Technical University of Munich, Germany
[5] Mace and Northumbria University, United Kingdom

artur.b.tomczak@ntnu.no

Abstract. With the ever-growing digitalisation of the built environment, specifying information requirements (IR) is crucial to control the Building Information Modelling (BIM) data. However, the way of encoding these requirements is subject to a wide range of possibilities, making it difficult for the users to choose the most suitable method. The paper presents a comparative analysis of methods to define IR based on document study and expert group discussions. The study covers Data Dictionaries (ISO12006), Information Delivery Manual (IDM), IFC Property templates, Information Delivery Specification (IDS), Level of Information Need, Model View Definition (mvdXML), and Product Data Templates (PDT), as well as other, non-standardised methods such as Linked Data with SHACL. The comparison is based on criteria from the review of use-cases and covers aspects such as value constraints, properties of fields, geometry representation, metadata, expressiveness and dependency. The paper concludes that no single method covers all the discussed aspects, and selection should be made consciously based on a purpose. The results are relevant to information managers to understand the relations and differences between IR methods, suggest standardisation bodies a way forward to harmonise, integrate or differentiate the standards, and provide a framework for evaluating IR methods.

Keywords: Building Information Modelling (BIM), Exchange Requirements, Information Management, Information Requirements, Model Compliance Checking

1. Introduction

The Architecture, Engineering, Construction (AEC) industry is highly fragmented and operates in projects with ad hoc teams of specialists and stakeholders. Data deficiency, inaccuracies or redundancy are common sources of inefficiencies and reduce trust in the data. To achieve effective collaboration and avoid disappointing results, project agreements and deliverables should be clearly defined.

Before the building is constructed, it is a non-tangible and abstract design. For all stakeholders to come to an agreement, they should share a common understanding of the concepts it entails. Building Information Modelling (BIM) is the predominant methodology for data exchange on medium and large design projects in many countries [1]. It combines geometrical and alphanumerical built asset information in the digital environment. Previous research shows that a significant barrier to BIM adoption is the challenge of formalising the information requirements (IR) [2].

The prevailing standard series ISO 19650 on information management using BIM, define information as ‘reinterpretable representation of data in a formalised manner suitable for communication, interpretation or processing by human or automatic means’, and IR as ‘specification for what, when, how and for whom information is to be produced’. In other words, IR is the request for explicit information to be delivered at a given time of the project to an indicated recipient, in a prescribed method and for a given purpose. The series organizes IR based on the context of the document: Organizational (OIR), Asset (AIR), Project (PIR), altogether providing input for the Exchange Information Requirements (EIR) that define concrete contents for individual exchange scenarios. IR are usually confirmed and further detailed in the BIM Execution Plans (BEP). Well-managed and structured IR contribute to more informed decisions, can enable data-driven workflows, and allows for process automation [3].

To create interoperable software capable of data exchange, vendors implementing BIM processes into their products rely on international standards, such as Industry Foundation Classes (IFC) which provides a comprehensive vendor-neutral data model for BIM. However, more than one standard pertains to IR, and despite the multiplicity of guidance on managing project information, there is no consensus on the method to represent and validate IR. In common practice, IR are also specified in custom documents or hard-coded programming solutions not following any standard. Such resemblance might lead to misuse of the solution, causing confusion and inefficiency. To prevent this, the paper compares the most popular methods of managing IR, both standardised and not. The intention is not to look at what should be required with IR, but how requirements themself should be defined, shared and validated against with a focus on technical implementation.

2. Literature Review

From the analysis of the standards, the authors have identified those that provide methodologies for IR: Data Dictionaries, Information Delivery Manual (IDM), IFC Property templates, Information Delivery Specification (IDS), Level of Information Need, Model View Definition (MVD), Product Data Templates (PDT). Another popular method that appeared in literature on Linked Data is Shapes Constraint Language – SHACL [4]. Existing literature on BIM includes the abovementioned methods, but rather in isolation, lacking an overview and comparison. Only two papers found in Scopus mention as much as three out of eight of the methods at a time, which can be justified given that many of the standards have been just recently published. MacPherson [5] addresses in the paper IDM, MVD and PDT, in the context of Web Services, that reuse existing standards, but without focusing on IR. The second paper, by Gigante-Barrera et al. [6] mentions IDM, MVD and Data Dictionaries focusing on content specification at various Levels of Development (LOD), but regardless of methods to define and validate such specifications. It is considered a knowledge gap worth addressing, given that such a holistic perspective can aid decision-makers in selecting a method to fit a particular purpose and improve efficiency and communication. The study also provides a framework for evaluating standards that define IR methods.

3. Methodology

Inspired by the comparative analysis guidance from Walk [7], the frame of reference of the paper was set to solutions to manage IR in the context of BIM and the construction industry. This implies that aspects of described methods falling outside the scope are omitted intentionally, and one should not judge evaluated methods regardless of their entire intended purpose. The primary focus is on quantitative IR, while qualitative IR, such as “a room should inspire creativity”, are excluded from this analysis. To identify initiatives, a review of standards and literature was performed. Grounds for comparison are defined based on our analysis of use-cases available in the Use Case Management

Service (UCM)[1], deliberately selected to describe the range of industry needs well. Then, the methods were analysed during 12 remote meetings of 1-2 hours each, supplemented with inputs from four external experts. The analysis from experts' discussion includes standards that are both published and under development as well as unstandardised methods and templates. Because it highly depends on the application, the paper focuses on the templates of BIM EIR prevailing in practice which typically associate (domain or project-specific) object types with required properties.

The organisational scheme of the body of the paper consists of three parts: (1) determination of the evaluation criteria, as a part of the methodology, (2) description of each method with a brief history and its intended purpose, and (3) a point-by-point comparison and evaluation based on the criteria, followed by discussion.

3.1. Evaluation Criteria

The UCM provides a reference to real use-cases on best practices and BIM workflows. From their analysis, the key aspects that will be taken into consideration in later analysis of each method were identified. Those are summarized in Table 1 and explained in the following paragraphs. The results present assessment on how each method performs in those aspects. However, due to some methods' generality and ambiguity, the evaluation is not definite. Hence, the resultant Table 2 only indicates the relative capability of each method, and Table 3 shows the grouping of similar methods according to given criteria. The dissimilarities and challenges with evaluation are explained in the text.

Table 1. Evaluation framework summary

Applicability	Fields					Value constraints				Content			Geo-metry		Metadata			Other			
	Information type	Data type	Unit of measure	Description	References	Equality	Range	Enumeration	Patterns	Existence	Documents	Structure	Representation	Detailedness	Purpose	Actors	Process map	Expressiveness	Dependency	Tech. agnosticity	Governance

It is common for IR to specify their applicability – assignment to which elements they apply to, for example, is it a wall, a pipe or else. Another prevailing aspect is the specification of the type of information that the requirement is querying, whether it is a property, material, or else. Some cases also require field values of certain data types, like text, number, boolean etc. Numerical values can also be defined with intended units of measure. Requirements meaning might be further explained with the plain text description, especially useful for unambiguous contractual definitions. The global community speak different languages, use other units and has other names for similar concepts. What IFC calls '*thermal transmittance*', the ETIM classification calls the '*U-value*'. While experts can interpret such ambiguities, machines and software developers can not, which is why references to a common standard are desired to help avoid misunderstandings.

For data to be reliable, it should undergo a validation process. According to EN 17412-1, validation is a 'confirmation, through the provision of objective evidence, that the requirements (...) have been fulfilled'. All of the analysed methods are considered machine-readable, but not in all can IR be automatically validated for delivered models without much human processing or only approximate Natural Language Processing. Thus, a clear distinction was made between machine-readable and machine-interpretable IR. The objective evidence can only be found if requirements are explicitly stated with value constraints. Alphanumeric data can be validated for equality against literal values, but it could be more advanced, allowing for multiple possibilities or synonyms (enumeration), or variance of notation validated using Regular Expression patterns. Numerical values might be limited

---

[1] https://ucm.buildingsmart.org/, accessed 02.03.2022

to defined range boundaries, as in the case of relative humidity, which is expected to be between 0 and 100 percentage units.

When no element matches a requirement, whether the model should pass such validation depends on how IR method specifies information existence. Propositional logic only deals with the truth value of propositions and logical connectives, but to fulfil use-cases needs, method should also support first-order logic, allowing to make statements with respect to a population by introducing universally (∀) and existentially (∃) quantified variables. For example, the universal requirement 'all walls need a *length* property' would be satisfied with the lack of any wall but fail if only one of many walls does not have a *length* property. In contrast, existential requirement 'a wall with a *length* property must exist' would result opposite: failure and success correspondingly. Requirement for a content existence can also extend to documents such as attached files, drawings and images. Furthermore, IR could be specifying how an element or data should be structured with relation to other objects. For example, is it enclosed in a larger assembly or contained within the spatial construct? Structure applies to both spatial composition and how the data is structured in the model hierarchy.

Requirements can also refer to spatial information. In many cases it is sufficient that geometry is described with boundary representation or mesh, but in some, it is needed that representation describes embedded logic, explaining how the volumes were procedurally extruded. An example of such requirement is found in the SIMBA[1] use case from the Norwegian Statsbygg, that demands that *IfcSpace* object extends from the upper surface of a floor slab to the underside of a slab above. IR can also demand a particular level of detailedness, meaning what needs to be modelled and to what precision. The more schematic model usually takes less memory and time to create, but a more granular one can provide more details necessary for the later design stages.

The context of IR is given by providing metadata. According to EN 17412-1, validation differs from general verification by having a specific intended purpose. Some use-cases need to specify an actor to assign the responsibility for fulfilling the IR to a person or role. Finally, use-cases might derive IR from workflows, so a method should support mapping IR to a process map with data exchanges.

Apart from the described features, other aspects are considered to give a holistic overview of the IR methods. These include the expressiveness of standards, meaning the breadth of ideas they can describe, and how ambiguous they are. The next aspect is dependency, both external – on industry classifications and schemas that define the semantics, and internal – on technology. The latter, referred to as technology agnosticity, includes the complexity of transcribing a standard from one file format or implementation to the other. The European Interoperability Framework discourages over-restrictive obligations to use specific digital technologies [8]. Finally, there are questions about the method's governance: how it is managed, is it standardized and transparent, is it coordinated, how easy it is to influence change.

4. Results

4.1. Description of IR specification methods

Today, the most common method for IR is text-based documents (DOC). They are usually authored with popular text editors, supplemented with images and diagrams, and compiled into PDF files. It owes its popularity to the versatility of such form and common user acquaintance with the tools. It is also grounded in the legacy of paper-based contractual documents, prepared by lawyers and managers often unfamiliar with BIM technicalities. A slightly more structured way of defining IR is using spreadsheets (XLS). It imposes a tabular layout and is easier for computer parsing.

One attempt to standardize a tabular set of product properties is with Product Data Templates (PDT), also called Data Templates. PDT was established to enable the comparison of similar products, mainly for the tendering phase of the construction project. Sharing the same templates allows for smooth communication between manufacturers and purchasers. Once the template is populated with data, it becomes the Product Data Sheet, preferably embedded in the information model. Besides public PDT collections, such as for the BREEAM certification, everyone can create their own PDT. While the properties themselves are not standardized, the methodology how they should be serialized, maintained and described with meta-data attributes are defined in ISO 23386.

ISO series 29481 brought Information Delivery Manual (IDM) to link business processes with the specification of information to be delivered to facilitate interoperability between software applications. While PDT’s focus is on 'what', IDM adds focus on 'by who', ‘for whom’, 'when' and 'for what'. It captures the responsibilities of each role to provide information at a specific time of the project, defined for example by its predecessors. IR – here called ‘Exchange Requirements’ – are an essential of the four integral parts of IDM, among interaction, transaction, and process maps, which utilize Business Process Modelling Notation (BPMN) to express purpose of data flows between people. Until recently, IDM was a formalization of text and tabular document with IR, not easily reusable due to lack of common standard data schema, but currently being developed part three of the standard is supposed to provide the standardized notation in form of idmXML schema definition.

According to the first part of the ISO 29481, the IR in IDM are the basis for creating the Model View Definitions (MVD) – the technical specification of the expected software implementation. The original purpose of MVD was to limit the scope of much broader IFC to just the subset of concepts relevant to support a particular use case, as defined in IDM. Using MVD, software developers can comply with the rest of the specific industry branch without implementing irrelevant parts of the entire schema. Moreover, MVD also allows extending the schema with new properties, and relations between concepts. MvdXML – the format to encode MVDs – includes the *ExchangeRequirement* element, which could be an XML tree specifying constraints that information should follow, or a string following dedicated grammar, defined in the external annex. Such a feature encourages users to define individual project MVDs and use for IR validation. However, since its main purpose is software implementation certification, it is not convenient for defining project or organization specific IR [9].

To separate the MVD function of software certification guidelines from customizable IR, buildingSMART has proposed the Information Delivery Specification (IDS). The main goal of IDS is to provide a simple yet comprehensive way to author and validate nongeometrical IR. The validation result shall be unequivocal regardless of implementation so that either a human or machine interprets it equally. Like IDM and MVD, IDS also utilises the XML markup language. Each IDS specification consists of two parts: *applicability – that filters the elements,* and *requirements – that* request information to be delivered. What can be asked for in IDS is limited. It can be a property, material, entity, predefined type or classification. How those are interpreted, rely strictly on the IFC schema.

To aid communication, the part three of the ISO 12006 defined the language-independent information model, which can be used to develop data dictionaries (hereinafter DD) used for storing information about construction works. It enables classification systems, information, object, and process models to be referenced within a common framework. DD define a standard structure to store objects with their properties and relations, addressing ISO 23386 need for meta-information attributes. This includes references to external documents and information about when the objects were created and by whom. An example implementation of this standard is the 'buildingSMART Data Dictionary' (bSDD)[2]. It provides a single-entry point for accessing multiple dictionaries through its application programming interface (API) and Uniform Resource Identifiers (URI).

The more details a model contains does not necessarily convey more quality information. A very precise but inaccurate value can lead to wrong assumptions at project appraisal. Similarly, very complex graphical detail might obscure the essential idea an architect wants to convey. For that reason, for years, BIM projects were applying the LOD (Levels of Development/Level of Detail), that specifies the accuracy of information usually related to geometrical modelling [10]. ISO 19650-1 introduced the Level of Information Need (for the paper abbreviated as LOIN), which is another approach to IR, further specified in the norm EN 17412-1. In contrast to the LOD specification

---

[2] http://bsdd.buildingsmart.org/, accessed 02.03.2022

proposed by BIMforum[3] and other industry guidance, the LOIN standard deliberately refrains from defining explicit levels. It starts by defining pre-requisites that must be specified to establish the context information: purpose (why) e.g. energy analysis or visualisation, information delivery milestone (when), the actor (who) and the object the requirement is associated to. The object can be an element (door, floor) but also a system or the entire asset; it depends on what we want to achieve. Then the standard defines the concepts and principles to specify geometrical information using five different aspects: detail, dimensionality, location, appearance and parametric behaviour. Beside geometrical needs, LOIN also allows formulation of alphanumeric IR and requirements for documentation that can be transferred in addition to the geometrical information. These can include, for example, construction drawings, product data sheets or construction schedules. Part 3 of the standard is under development and will provide a data schema based on EN 17412-1.

IFC version 4 introduced the concept of IfcPropertyTemplate (IPT)[4]**.** It is a place in the specification for defining templates for creating property set instances. IPT is not a straightforward solution for requiring information but can specify how information should be delivered in an IFC model. IPT allows a field to be of any value, but it can also bound it to certain limits, enumeration, list, reference, or a table. It can also limit a property to represent a predefined quantity such as length, area or volume, preventing mistakes such as negative values. An example of IPT could be a template for scalar property for ranking that only allow integers between 1 and 5. In current practice IPTs are being used to define IFC property sets that belong to manufactured products. Even though it is the only method embedded in the IFC specification, it is not commonly used in practice for IR.

In addition to textual and standardized methods, the common approach to IR in the industry is to use proprietary software solutions – hereinafter 'Other’. There are commercial software applications that allow IR authoring and even contain predefined sets of IR and rules to check building models. Some companies build their own tools, for example, employing visual programming, which gives freedom of definition but is tied to hardcoded functions. Others overcome the limitations by developing open-source code solutions, such as with higher-level human-readable scenarios written in Gherkin syntax [11].

Another option is to transform the IFC model to a more queryable form such as a graph database or use specific purpose querying languages [12-13]. In particular, the usage of IfcOWL [14] comes to mind as the Linked Data (LD) and Semantic Web community has a wide range of standards for reasoning and validation. This paper uses Shapes Constraint Language (SHACL) as an example of a system to validate requirements on LD transformations of IFC. Oraskari et al. [4] discusses the conceptual similarities between MVD and SHACL. SHACL allows both filtering, validation and inference of data sets using arbitrary graph patterns, and constraints can be applied as SPARQL filters and XSD facets. Due to limitations in page count the exact formulation of shapes is not elaborated upon.

4.2. Comparison

The DOC and XLS documents templates are governed solely by file authors, usually, the people involved in the project or corporations. Proprietary software allegedly gives freedom of defining all workflows and features one might want, but it is tied to vendors intellectual property. Driven by user needs, a software developer can adjust the software or database, but its validation logic remains a black box to other users. IDM, IFC (IPT), PDT, DD, IDS and LOIN are either standardized by ISO or waiting to be. This gives them credibility and compatibility with other standards such as ISO 12006-3 and ISO 19650 series. The drawback is that standardization is a long process. The buildingSMART organization governs the MVD, IDS, IFC (IPT) and DD instance – bSDD, shortening the feedback loops by releasing intermediate versions.

---

[3] https://bimforum.org/LOD, accessed 02.03.2022

[4] http://standards.buildingsmart.org/IFC/DEV/IFC4_3/RC2/HTML/schema/ifckernel/lexical/ifcpropertytemplate.htm, accessed 02.03.2022

Next feature – the possibility of proper selection of applicable elements – is not in scope for DD, while LD+SHACL and Other are almost unlimited with this regard. For remaining methods, the possibilities range from not systemized plain text explanation of applicability criteria in DOC, XLS, and IDM, over filters based on object types in LOIN, to very precise filtering in MVD and IDS, in which one can filter the elements by IFC entity name, predefined type, element's property values, material or classification. In between qualify PDT and IPT, limited to only selecting individual entities.

Support for specifying field attributes is presented along other aspects in Table 2. Nonstandarized solutions, apart from spreadsheets, are excluded from the comparison as they rely heavily on particular implementation. Although all methods are capable of providing a field name, only DD, MVD, LD+SHACL and IDS can distinct information types: whether it is a property, IFC attribute, material association or external classification. MVD and LD+SHACL go a step further, allowing for any arbitrary type definition in graph patterns. Fields values can be restricted to particular data types in all four, as well as in PDT and IPT. Remaining standards have limited support for it. Some cells were left blank if the support is unknown while writing this paper.

Comprehensive descriptions are important part of DD and IDM, while all other methods support plain text explanations to some extent. References to common understanding are not possible in MVD and IPT. IDS, PDT, DD and LD+SHACL support references to norms, dictionaries and websites, although IDS is limited to one reference only.

Data constraint could be described with custom columns in XLS, but unless standardized it lacks the performance, so it is not considered supported. The focus of PDT is on providing data placeholders rather than requiring values. It also allows for any arbitrary definition, which might generate issues in automation and mapping of data from internal systems to a standard. IDS and SHACL rules in LD allow for constraining values with regular expression patterns (regex), crucial for example for requesting a prefix, upper or lower case, or particular number of characters. Although ISO 12006-3 does not support this, the bSDD implementation does have the XSD restriction capabilities like patterns and enumerations implemented similarly to IDS. While in mvdXML one can emulate enumerations using boolean or conjunctions, an explicit enumeration construct is missing. The mvdXML 1.1 dedicated rule grammar that is used to formulate constraints has a provision for specifying regular expressions, but the definition itself is fully opaque as simply a string without a specification of the supported constructs. Draft version 1.2 of the mvdXML proposes POSIX flavour to clarify the rules of regular expressions.

The existence, whether at least one object matching the criteria is present, can be checked with IDS' *use* feature (*required* or *optional*), as well as mvdXML's *requirement* (*mandatory*, *recommended*, *not-relevant*, *not-recommended*, *excluded*). Theoretically nonexistence cannot be confirmed in LD due to the open world assumption, but for practical purposes SHACL can validate presence of an element. Attachments, drawings, images and other documents are only anticipated in IDM and LOIN; however, LD+SHACL references might serve such a purpose. Spatial decomposition, whether object is part of a bigger assembly or hierarchical structure exceeds the scope of most methods. It can be required with MVD and LD+SHACL, again, thanks to their graph patterns. IDS can specify materials that an object is composed of and whether it is a part of a group, an assembly or a system. Partial support for spatial decomposition is seen in methods with plain text explanation, due to their expressiveness.

MVD and LD+SHACL can both express the type of geometry representation well, but not the detailedness. The IDM offers this, but the method offering most comprehensive detailedness of geometrical IR is present in LOIN, with the clear distinction of aspects such as detail, dimensionality, location, appearance, and parametric behaviour. Geometry is not in scope of any other method than this four.

Not all IR can be verified without additional context. Such an example is the requirement for Level of Development (LOD) that an element needs to be in its final location at a specific point in design development. It can be verified whether the value is 200, 300 or 400, but it cannot be automatically validated if the state would not change since the computer is only aware of the current location, and whether it changes is up to the designer. The best at providing broader context in the form of metadata is IDM. It has workflow and process maps linked with individual IR, for example, allowing specification of preconditions to be completed before each IR. Context of a purpose or milestone can also be given to IR in LOIN and IDS. Even though IDS is made for reliable data exchange workflows, it refers but does not define them.

Table 2. Information requirement support in the described methods.

◌ – No ◑ – Partial ● – Yes * – under development			Fields					Value constraints				Content			Geom.		Metadata		
	Standardised	Applicability	Info. type	Data type	Unit of meas.	Description	References	Equality	Range	Enumeration	Patterns	Existence	Documents	Structure	Representation	Detailedness	Purpose	Actors	Process map
Spreadsheet	◌	◑	◑	◑	◑	◑	◑	◑	◌	◌	◌	◌	◌	◌	◌	◌	◑	◌	◌
PDT*	●	◑	◑	●	●	◑	●	◌	◑	●	◌	◑	◌	◌	◌	◌	◑	◌	◌
Data Dict.	●	◌	●	●	◑	●	●	●	◑	●	◑	◌	◌	◌	◌	◌	◌	◌	◌
IDS*	●	●	●	●	●	◑	◑	●	●	●	●	●	◌	◑	◌	◌	●	◑	◌
mvdXML	●	●	●	●	●	◑	◌	●	●	◑	◑	◑	◌	●	●	◌	◌	◑	◌
idmXML	●	◑	◑	◑	◑	●							●		◑	◑	●	●	●
LOIN*	●	◑	◑	●	●	◑	◑	◌	◌	●	◌	◌	●	◑	◑	●	●	●	◌
IFC P.T.	●	◑	◑	●	●	◑	◌	◌	◌	●	◌	◌	◌	◌	◌	◌	◌	◌	◌
LD+SHACL	◌	●	●	●	●	◑	●	●	●	●	●	◑	◑	●	●	◌	◑	◑	◌

Remaining aspects are not applicable for grading, instead they will be grouped and presented in Table 3. In terms of expressiveness, plain text is the most unconstrained approach. On the other hand, a lack of rules opens such requirements for ambiguous interpretations. Ambiguity and the need for manual validation, might lead to oversight and inefficiencies. Slightly limited, but still quite expressive are IDM and MVD, since they base IR on language grammar. Available examples, such as abovementioned SIMBA[1], show that even though mvdXML is capable of more advanced definition, requirements are often expressed there with plain text. For that reason, it is assigned to both ends of the spectrum. IPT is rather strict, leaving no room for ambiguity, as IR needs to fully comply with IFC schema constraints. IDS is rather unequivocal but limits what can be expressed to properties, materials and classifications. DD also have limited scope of what can be expressed. As the data schema of LOIN is under development, its functionalities cannot be fully discussed and compared. Tabular solutions (XLS, PDT) are less limiting but usually rely on language grammar.

Interoperability depends on shared understanding. Several methods do not really convey any shared understanding, which is most evident in DOC. Spreadsheets define a repeatable structure for various definitions in a specification. However, the semantics of what a row/column/cell constitutes is not formal in the specification and depends on the data structure and conventions used for that particular sheet. Various standards rely on the schema to define the semantics of the specification. For example, an IDS specification might contain a *material* facet that may constrain the valid names for material, but the exact interpretation is defined external to the specification in the schema or its documentation. Lastly, mvdXML and LD+SHACL define both the semantics and the specification integrally. In the case of LD+SHACL, both schema and instance statements are triples with consistent formalization and inference rules. In the case of mvdXML, there is a listing of concepts, which are arbitrary IFC instance graphs, and then template rules to constrain such graph patterns with values ranges or patterns. This reduces the dependency on an external schema for interpretation and creates the flexibility to define arbitrary concepts in a specification but also removes the possibility for higher levels of abstraction. For example, IFC schema versions rename, remove or introduce attributes. Hence, the graph patterns for different schema versions are not compatible. This is visible in the SIMBA[1] MVD example, where a tight link to the IFC version requires creation of separate specifications.

Standards covered by this research do not dictate the technology that needs to be used. Some are usually plain text documents (DOC), some are defined as tabular (XLS, PDT, DD), and some are defined as XML schema – XSD (IDM, IDS, MVD and IPT), while LOIN schema is still under development. This does not preclude them to be transformed to other representations of data, such as relational databases or the internal data model of an authoring tool. However, the LD+SHACL approach realistically does require that the data model of the application is also in form of LD. This limits the ease and feasibility of embedding such IR checks in a wide variety of platforms. It also means that the set of specific problems of such technologies, in case of LD the inefficiencies in handling ordered lists [15], needs to be carefully considered.

Table 3. Grouping of information requirement methods.

<table>
<tr><td rowspan="2">Expressiveness</td><td>Free</td><td>Partially limited</td><td colspan="2">Very constrained</td><td>Strict</td></tr>
<tr><td>*DOC, IDM, MVD^a, Other^a*</td><td>*XLS, PDT, LD+SHACL, LOIN*</td><td colspan="2">*IDS, DD*</td><td>*MVD^a, IPT, Other^a*</td></tr>
<tr><td rowspan="3">Semantic dependency</td><td rowspan="2">Implicit</td><td colspan="3">Schema dependent</td><td rowspan="2">Self-contained</td></tr>
<tr><td>Indefinite</td><td colspan="2">Explicit</td></tr>
<tr><td>*DOC, XLS, MVD^a*</td><td>*IDM, LOIN, DD, PDT*</td><td colspan="2">*IDS, IPT*</td><td>*MVD^a, LD+SHACL*</td></tr>
<tr><td rowspan="3">Technology agnosticity</td><td colspan="4">Agnostic</td><td rowspan="2">Impose technology</td></tr>
<tr><td>Plain text</td><td>Tabular</td><td colspan="2">XML schema (XSD)</td></tr>
<tr><td>*DOC*</td><td>*XLS, DD, PDT*</td><td colspan="2">*IDM, IDS, MVD, IPT, LOIN*</td><td>*LD+SHACL, Other*</td></tr>
<tr><td rowspan="2">Governance</td><td>Custom</td><td colspan="2">buildingSMART</td><td colspan="2">ISO/CEN</td></tr>
<tr><td>*DOC, XLS, LD+SHACL, Other*</td><td>*MVD*</td><td colspan="2">IDS, IPT, DD</td><td>*LOIN, IDM, PDT*</td></tr>
</table>

[a] plural assignment, dependent on the implementation.

5. Discussion and conclusions

In this paper, several initiatives and workflows have been compared. The results show that no single solution covers all the aspects from the analysis of use-cases. Each method has a different intended purpose despite overlaps and should be selected consciously. All described methods have reasons for existence and play an important role in the ecosystem. However, because their boundaries are hazy, they tend to be misused, causing disappointment and limiting digitalization performance. The key differentiators identified by this research are as follows.

Data dictionaries and PDTs are best fit when properties are specified at a global level, shared with industry, not project-specific. They can also indirectly be used as a reference when defining requirements in all other methods to provide a shared understanding. Since PDTs do not restrict how to reference to norms and other values, there is a risk that equal content will be defined and referred differently, interrupting the interoperability and automation. The bSDD has mitigated this by restricting the units, references and country codes to a common list. LOIN is better suited on individual projects since it includes how information matures during the design, only asking for what is needed at a certain point in time. It also serves well for contractual deliverables, including geometrical requirements and documentation. However, LOIN's schema – EN 17412-3 – is still under development and not available for evaluation. The IDM has a key advantage of linking data handovers to exact processes and workflows. Its extension – idmXML – defines requirements in a computer-readable but not necessarily computer-interpretable way. For technical interpretation of human-readable instructions, it refers to MVD definitions.

Computer interpretability comes at the expense of expressiveness. The *IfcPropertyTemplate* is the most direct form of preparing placeholders for properties within the IFC schema, embedded in the schema itself. The same fact is the reason it is rarely used as a separate document, is limited to just basic features and does not allow for validation if elements are present. Despite being suggested by IDM, the MVD is primarily made for software certification rather than IR purposes. Its complexity stems from the ability to both define concepts with IFC instance graphs and constrain them. MVD has the advantage of being self-contained and able to define new concepts. However, being a bespoke, custom-made solution makes the software implementation only specific for supporting mvdXML, and readily available software libraries are likely not available. The complete picture can be drawn after considering their existing implementations. It was found that both mvdXML and IPT to not be widely adopted and hard to author in current tooling.

IDS has been identified as the most advantageous method, when it comes to automated compliance checking by validation of the alphanumerical IR. It supports IR authoring by providing users with a set of possibilities on what can be required of the models. On the other hand, unlike IDM and LOIN, IDS is just limited to IFC entities and their properties, making it only applicable to openBIM compliant tools. IDS is not made for geometrical requirements or mapping requirements with project processes.

Unstandardized methods, such as DOC, XLS, LD+SHACL and Others, overlap with all of the above, overcoming some of their limitations and adding flexibility both in definition and time of implementation. However, they also come with a tradeoff of limited transparency and integration with the entire ecosystem. The plain text allows describing all possible aspects but is loose and depend on the interpretation. LD+SHACL and Others require specialist knowledge to modify. In light of this knowledge, the question arises how much a standardized approach improves the collaborative workflow, reduces risks of errors and misunderstandings, and allows for optimization.

There are clear advantages and disadvantages of each method. The legal advantage of using text has a trade-off for automated compliance checking. However, automatic checking limits flexibility. This study should benefit those interested in specifying IR on BIM projects to make informed selection of the method and researchers seeking answers on challenges of BIM implementation.

Acknowledgments

The authors would like to thank the domain experts on idmXML, mvdXML and PDT who provided the material for description and evaluation of the standards. Special thanks to Jakob Beetz, Ghang Lee, Frédéric Grand, Matthias Weise, Eilif Hjelseth, Ole Jonny Klakegg, Anders Rekve, and Jardar Lohne for constructive feedback.

The potential source of bias comes from the involvement of some of the authors and contributors in standardization procedures and implementation processes, which could have affected the choice of use-cases and evaluation criteria. There might also be more methods to IR than were found in this study, due to their proprietary character or the localized application. To limit the bias, experts were selected that represent different perspectives, nationalities, and organizations.

The review of LOIN, IDM and PDT was hindered by the lack of a fully published version of the standards, making the results prone to minor change in the future. Similarly, MVD and IDS were evolving at the time of writing the paper. The difference in scope and intentions made the comparison non-trivial. Aspects such as the popularity of each method, state of their implementation, existing legacy, and the variance in standard execution between implementations were not included in the study. They would be worth following in future research.

References

[1] Kassem M and Succar B 2017 Macro BIM adoption: Comparative market analysis *Autom. Constr.* **81** pp 286–99

[2] Cavka H B, Staub-French S and Poirier E A 2017 Developing owner information requirements for BIM-enabled project delivery and asset management *Autom. Constr.* **83** pp 169–83

[3] Bolpagni M and Hooper E 2020 Guidance Part D Developing information requirements *Info. Manag. acc. to BS EN ISO 19650. UK BIM Framework https://www.ukbimframework.org/standards-guidance/* (accessed Jun. 28, 2021)

[4] Oraskari J, Senthilvel M and Beetz J 2021 SHACL is for LBD what mvdXML is for IFC *Proc. 38th Int. Conf. CIB W78* pp 693–702

[5] MacPherson D L 2012 Web Services and Ontologies for Building Data *Proc. 3rd Int. Conf. on Complexity, Informatics and Cybernetics, IMCIC* pp 33–5

[6] Gigante-Barrera A, Ruikar D, Crunden M and Ruikar K 2017 LOD object content specification for manufacturers within the UK using the IDM standard *J. Inf. Tech. Constr.* **22** pp 81–103

[7] Walk K 1998 How to Write a Comparative Analysis *Writing Center at Harvard University https://writingcenter.fas.harvard.edu/pages/how-write-comparative-analysis (accessed Dec. 30, 2021).*

[8] European Union 2017 New European Interoperability Framework *Publications Office of the European Union* https://ec.europa.eu/isa2/sites/default/files/eif_brochure_final.pdf (accessed Jun. 28, 2021)

[9] Berlo L V 2019 The curious case of the MVD https://blog.buildingsmart.org/blog/the-curious-case-of-the-mvd (accessed Jun. 28, 2021)

[10] Bolpagni M and Ciribini A 2016 The Information Modeling and the Progression of Data-Driven Projects *Proc. of the CIB World Building Congress* pp 296–307

[11] Moult D and Krijnen T 2020 Compliance checking on building models with the Gherkin language and Continuous Integration *Proc. EG-ICE Work. Intell. Comput. Eng.* pp 294–303

[12] Daum S and Borrmann A 2014 *Proc. Topolog. BIM Queries using Bound. Repr. Based Methods Adv. Eng. Informatics* **28** 4 pp 272–86

[13] Krijnen T and Beetz J 2020 An efficient binary storage format for IFC building models using HDF5 hierarchical data format *Autom. Constr.* **113**

[14] Pauwels P and Terkaj W 2016 EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology *Autom. Constr.* **63** pp 100–33

[15] Pauwels P, Krijnen T, Terkaj W and Beetz J 2017 Enhancing the ifcOWL ontology with an alternative representation for geometric data *Autom. Constr.* **80** pp 77–94