MVD policy for IFC 4.x

Date: April 2021

Status: Published

Version: 0.3

Contents

Executive summary

Model View Definitions (MVDs) are buildingSMART's solution to create IFC based standards that can be implemented and tested.

An MVD consists of three main components:

- A set of Concept Templates. These concept templates define additional agreements on how to use the IFC Schema.
- A set of Exchange Requirements. This is a selection of entities and properties from the IFC Schema that are found suitable for a selection of use-cases.
- A description on how Software should deal with the data that are exchanged. For example, can the software use the data as a reference, or should the data be mapped to internal objects during import.

The buildingSMART Software Certification team operating the b-Cert platform (b-cert.org) is using MVDs to test software implementations for conformance.

In the buildingSMART community the term MVD has been mainly used to define exchange requirements in a project. Within the buildingSMART solutions this is however covered with the combination of IDM and IDS.

In principle, everyone can define their own MVD. Without common guidelines it is unlikely that these MVDs will be interoperable with other MVDs and likely that they will need specific software implementations. MVDs become non-interoperable when different resources or concept templates are being used. This is described several times[1] and is already happening in current practise. The objective of these guidelines is to set out a framework to increase interoperability between MVDs and reduce implementation, testing and certification costs.

Creating an MVD that is not interoperable with other MVDs creates the following situation:

- Software vendors that implemented an MVD most likely need to make a separate effort to implement another MVD.
- Data that is being exported according to one MVD cannot be read by a software tool that only implemented the other MVD.
- Projects that have defined their MVD cannot use already existing software implementations.

To avoid these situations buildingSMART is defining a limited amount of 'base MVDs'. These base MVDs can be extended with custom made sets of exchange requirements but cannot change concept templates and the selection of IFC resources.

This policy supports projects with delivering valid and interoperable exchange requirements for their use-cases, it provides interoperability with existing software implementations, and helps align with Software Certification.

---

[1] A comprehensive overview is on https://blog.buildingsmart.org/blog/the-curious-case-of-the-mvd

1   Background & situation

At the time of writing this policy, there is a formal IFC 4 release, with a Reference View MVD version 1.2[2].

IFC 4.x versions 4.1 and 4.2 are withdrawn and IFC 4.3 has three release candidates[3]. There are several draft MVDs for IFC4 and IFC 4.2. The IFC Bridge MVDs 'Bridge Reference View', 'Alignment Based Bridge Reference View' and 'Bridge Design Transfer view' are notable draft publications.

The IFC Rail and IFC Infra projects working on IFC 4.3 have listed 'Reference View', 'Alignment Based Reference View', 'Design Transfer View' and 'Asset Management handover' as potential MVDs for IFC 4.3.

The IFC Tunnel project has similar MVD ideas for IFC 4.3.x.

In the USA, there is an initiative to create USA based exchange requirements on top of 'Alignment Based Reference View' base that is intended for the Infra/Rail projects.

Every IFC version has a 'General Usage' MVD with a set of Concept Templates. These are seen as part of the IFC standard.

Several non-infra projects are creating exchange requirements for Software Certification, or even separate MVDs. This document is intended to guide projects to efficient choices, applicability in existing software implementations and a strong connection to potential software certification.

1.1 Scope

Currently there is a Reference View for IFC 4 and an upcoming release of IFC 4.3. This document has a focus on defining a policy for MVDs in IFC 4.x. In practise that means IFC 4.3.x with backward compatibility to the Reference View of IFC 4.

Although this document has a focus on the Infra/Rail MVDs it is not limited to those and also applies to building or building services related MVDs and exchange requirements.

1.2 Objective

The objectives of this policy are:

-   To support projects making decisions on defining exchange requirements and using concept templates;
-   To create interoperability between new exchange requirements and existing implementations of MVDs in software tools;
-   To create interoperability between IFC files that are being exchanged in practise;
-   To support projects to provide the necessary information to the b-Cert team to start Software Certification of their MVDs;
-   To help the b-Cert team to start Software Certification of IFC 4.3.x MVDs as soon as possible after formal release of IFC by buildingSMART.

---

[2] https://standards.buildingsmart.org/MVD/RELEASE/IFC4/ADD2_TC1/RV1_2/
[3] https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/

1.3 Future of MVDs

Currently everyone can create their own MVD, deviating from the common buildingSMART MVDs. This is not a desired situation for the future[4].

In IFC 5 the concept templates will be a more integrated part of the IFC standard, and additional exchange requirements can be defined in a flexible way.

This will also impact future Software Certification.

---

[4] https://www.buildingsmart.org/wp-content/uploads/2020/09/20200430_buildingSMART_Technical_Roadmap.pdf

2   Different components in an MVD

It seems the topic of MVDs has several misconceptions in the buildingSMART community. For reference purposes, this chapter provides a short and simplified overview of the several components inside an MVD.

Figure 1 provides an overview of the layers of the most common MVDs for IFC 4.x
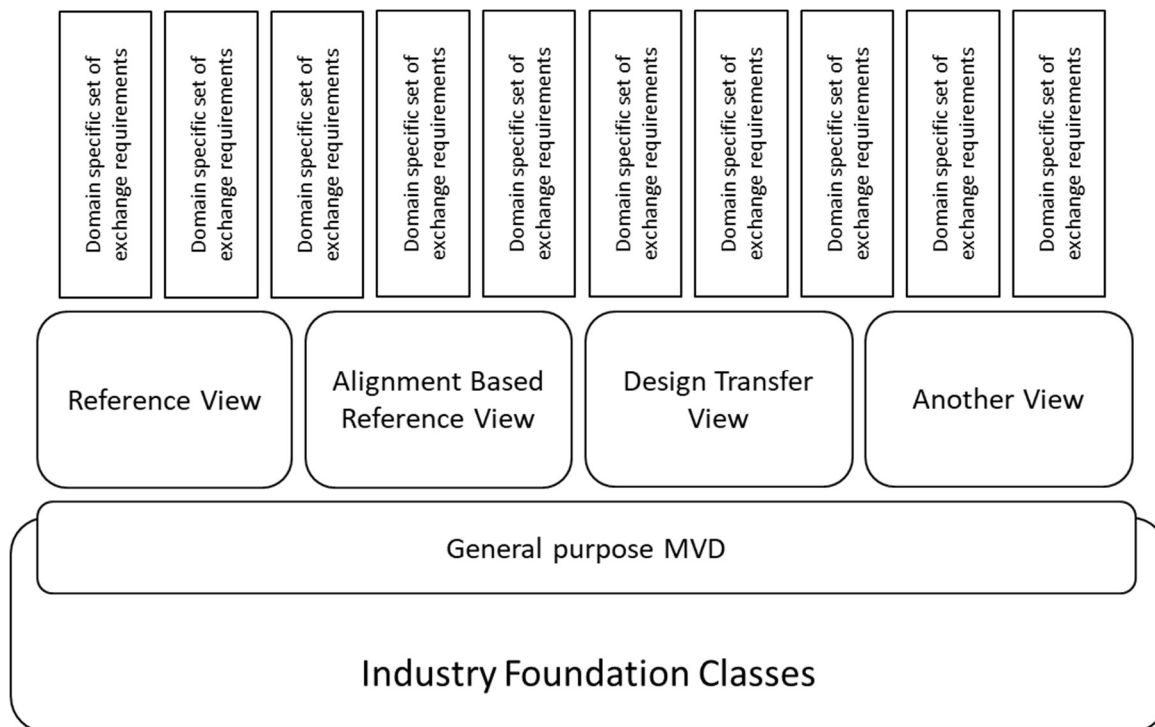

Figure 1: Components in an MVD

At the bottom there is IFC. IFC is defining entities, properties, predefined types, predefined property sets, resources like materials, geometry, units, and many more things. These semantic agreements are a vital standard reference for many use-cases.

The standard needs to define how properties can be connected to objects, or how materials can be connected to entities. There also needs to be restrictions on how to use the spatial structure, how to define colour to geometry, etc. This is done in so-called 'concept templates'. Many of the most common things are defined in the 'general usage' concept templates. This is part of the formal release of IFC.

A selection of the IFC standard can be made for a specific purpose. This is called an MVD and is currently often described in an mvdXML. The first part of an mvdXML file is the definition of concept templates. Creators of an MVD are not limited to use existing concept templates from 'general use' but can create any kind of concept template themselves.

On top of the concept templates, there are often one or more sets of so-called 'exchange requirements'. It is worth noting that the <ExchangeRequirements> node in an mvdXML file is something other than the 'Exchange Requirements' term as it is generally used in IDM and ISO 19650. IDM and ISO 19650 define the term 'exchange requirements' as a 'human readable' list of what is needed in a project. However, the terminology of 'Exchange Requirements' in an MVD precedes the IDM and ISO 19650 definition.

One MVD can have multiple Exchange Requirement sets. For example, the Reference View of IFC 4 has 'Architectural Exchange', 'Structural Exchange' and, 'Building Services Exchange'.

The selected entities and properties in an Exchange Requirement can only have resources that are part of the concept templates. The set of exchange requirements can be the same for a different set of concept templates.

Additionally, to this overview, an MVD also defines requirements as to how software shall deal with the data internally. This is often done with human readable documentation. This 'conformance level' can for example define if IFC data can be used as a reference inside the software, or that is should be mapped to internal objects. Although these requirements are often also part of an IDM, this needs to be part of the MVD as well when Software Certification is desired.

2.1 b-Cert Software Certification

Currently the b-Cert team that is executing Software Certification is heavily reliant on MVDs and mvdXML.

The team is testing the concept templates, and the minimum needed set of Exchange Requirements.

Figure 2: Screenshot of b-Cert Test Center

As can be seen in Figure 2, the b-Cert Test Center is testing IFC files based on the different Exchange Requirement filters, as well as the Concept Templates defined in the MVD.

3   Interoperability between MVDs

In principle, everyone can define their own MVD. There is a risk, even likelihood, that MVDs are being defined that need specific software implementations because they are not interoperable with other MVDs. MVDs become non-interoperable when different resources or concept templates are being used. This has been described several times[5] and is already happening in current practise with IFC4 and clearly this is not a good thing introducing complexity, waste and additional costs.

Creating an MVD that is not interoperable with other MVDs creates the following situation:

- Software vendors that implemented an MVD most likely need to make a separate effort to implement another MVD.
- Data that is being exported according to one MVD cannot be read by a software tool that only implemented the other MVD.
- Projects that have defined their MVD cannot use already existing software implementations.

To avoid these situations buildingSMART this policy is defining a limited amount of 'base MVDs'. These base MVDs can be extended with custom sets of exchange requirements. Projects are strongly discouraged to create new concept templates or make changes in the selection of IFC resources.

Base MVDs do not have extensive exchange requirements. Projects that want to define exchange requirements do that as an extension to the base MVDs.

This policy supports projects with delivering valid and interoperable exchange requirements for their use-cases, it provides interoperability with existing software implementations, and helps align with Software Certification.

3.1   The base MVDs in IFC 4.x
Based on the input from the IFC Infra project, IFC Rail project, IFC Tunnel project, and the USA project for AASHTO, the following MVDs are proposed:

- **Reference View**
- **Alignment Based Reference view (infra)**
- **Design Transfer view**
- Optionally: Asset Management Handover

The Reference View already exists in IFC 4, and the Reference View in IFC 4.3.x should be backward compatible with the existing one. The alignment-based reference view will be specific for use-cases that need an alignment-based geometry. The Design Transfer View has never been matured and released but is still on the 'most wanted' list of projects and stakeholders.

The Asset Management Handover MVD seems to be a set of 'Exchange Requirements' that can be based on simple geometry from Reference View, or in some cases even without geometry. Therefore the 'Asset Management Handover' will be considered an 'Exchange Requirement' set on top of the Reference View until new insights might reveal it should be a separate one.

---

[5] A comprehensive overview is on https://blog.buildingsmart.org/blog/the-curious-case-of-the-mvd

4   Project deliverables for MVDs

It is strongly advised that projects in buildingSMART Rooms use one of the defined MVDs and extend[6] and refine that MVD with their own set of Exchange Requirements.

Exchange Requirements are often domain specific and provide the Software Certification team input for testing compliance of implementations. Therefore, the definition of Exchange Requirements on top of one of the base MVDs should be done in close collaboration with the b-Cert team.

When projects are defining 'exchange requirements' that do not need Software Certification there are many different alternatives to do so.

Since not all MVDs are defined yet, there is an opportunity for projects to collaborate with the b-Cert team. The opportunities and restrictions for the definition of MVDs are defined in the following paragraphs.

4.1   Reference View

The Reference view is the most stable and mature MVD of the IFC 4 family. For IFC 4 the Reference View has version 1.2 as the most recent one.

For IFC 4.3 the Reference View needs to be updated to comply with some new terms and minor changes in IFC 4. To keep backward compatibility between Reference View in IFC 4.3 and IFC 4, the agreements are fixed and cannot be changed.

4.2   Alignment Based Reference View

The alignment-based Reference View was first introduced by the IFC Bridge project for IFC 4.2. Since the Alignment in IFC 4.3 is different from IFC 4.2 and IFC 4.1, the IFC 4.1 and IFC 4.2 releases have been withdrawn.

The alignment-based Reference View from Bridge (in IFC 4.2) can be a starting point for the alignment-based Reference View in IFC 4.3 (and later).

With the significant changes to Alignment, the update of alignment-based Reference View from IFC 4.2 (bridge) to a base MVD for IFC 4.3 could be a challenge.

---

[6] Extend means "Adding a set of exchange requirements on top of the base concept templates"
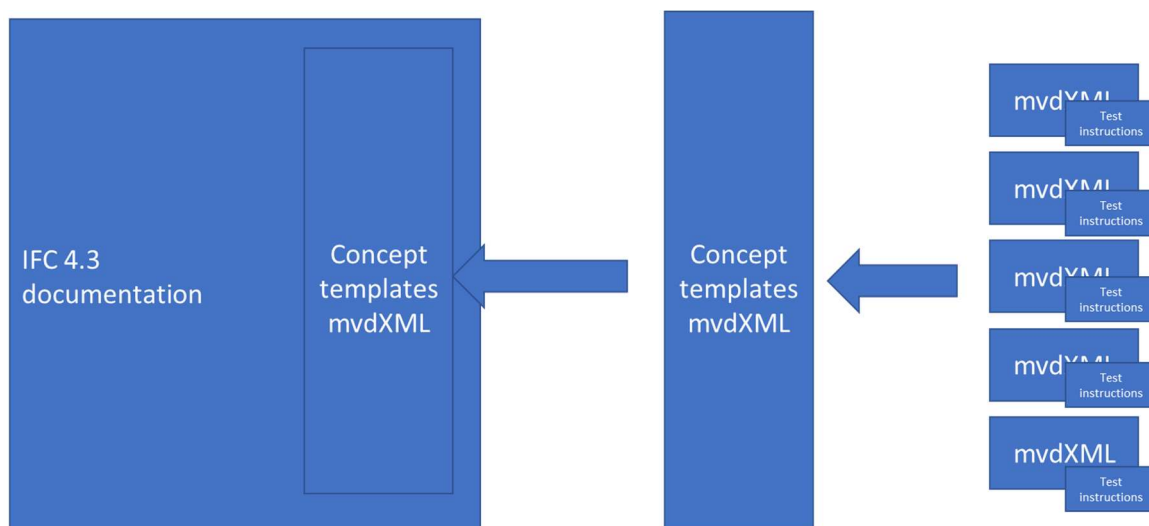
Figure 3: workflow for defining alignment-based reference view for IFC 4.3.x

Projects that want to define the alignment-based Reference view can get support from the Software Certification Team of b-Cert.

The test instructions are important to define together with the b-Cert team, since these are instrumental to high quality Software Certification.

Base MVDs do not have extensive exchange requirements. Projects that want to define exchange requirements do that as an extension to the base MVDs. Exchange requirements can be shared with b-Cert as described in chapter 4.4.

4.3   Design Transfer View

The Design Transfer View has never been defined properly.

Projects that want to define the Design Transfer view need to follow the same procedure as for the Alignment Based Reference view (see Figure 3).

4.4   b-Cert ready Exchange Requirements

Projects that want to add an addition set of Exchange Requirements as an extension of the mentioned 'base MVDs' need to contact the b-Cert team. This is only applicable when Software Certification is desired.

Exchange Requirements can be delivered to the b-Cert team in mvdXML, but also in excel or any other deliverable that is agreed with b-Cert.

Domain specific exchange requirements also need detailed test instructions to guarantee a valid procedure for Software Certification.

4.5   General comments

It is important to realize:

- Concept templates can only change after a new IFC 4.x release.
- Although some concept templates or conformance in Software is not defined properly yet (for example Design Transfer View), this does not mean project have full freedom. The base MVDs are following a certain purpose and vision that is described by multiple projects.

- The b-Cert platform is publishing resulting MVDs (including formal defined exchange requirements) as mvdXML on GitHub[7].
- IFC 4.x base MVDs can only be released after being created and verified on the b-Cert platform.
- Domain specific exchange requirements on top of 'base MVD' concept templates, can only be released after approval by b-Cert. b-Cert can only approve exchange requirements when they are accompanied with specific test instructions.

---

[7] On https://github.com/buildingSMART/IFC4.3.x-development/tree/master/mvdXML

5   Exchange Requirement MVDs

Many MVDs that have been made over the years, are focussed on defining the exchange requirements. Examples are the FM Handover and Quantity take-off MVD.

Since these MVDs are exchange requirements that can work on any base MVD, these can be used as such. They can also be used on top of 'bare' IFC ('general usage' IFC).

Almost every of these MVDs has a special story. The FM Handover for example comes with an additional excel based file format.

In future, the 'Information Delivery Specification' standard is likely to be used for defining use-case specific exchange requirements. At this phase, the exchange requirements (see Figure 1) can work on any base MVD or on general usage IFC.

5   Exchange Requirement MVDs

Appendix: Proposed workflow for IFC 4.3.x projects

This policy was approved by the SCTE in the January 2021 meeting. At that time there were several projects working on the IFC 4.3.x release.

The following workflow is proposed for the projects to help them define MVDs with Exchange Requirements as reliable and efficient as possible:

1. Update Reference View from IFC 4 to IFC 4.3.x.
2. Update Bridge MVD(xml) to create alignment based Reference View.
3. Finalise IFC 4.3 Reverence View.
4. Update documentation on Reference View and alignment-based Reference View.
5. Optional: create Design Transfer View concept templates and documentation
6. Provide exchange requirements per domain (suggested to use IFC domains + bridge).
7. Provide test instructions for b-Cert to test software.
8. Publication in b-Cert will automatically be pushed to GitHub and become part of the formal deployment chain.

This flow needs to be executed in close collaboration between the Infra/Rail projects and the b-Cert team. Potentially the AASHTO project and IFC Tunnel can also join.

Appendix: Terminology

bSDD	buildingSMART Data Dictionary	An online service from buildingSMART International to host and distribute data standards.
b-Cert	buildingSMART Software Certification platform	Available on b-cert.org. This platform is used by software vendors and the buildingSMART Software Certification team to test the MVD implementations of software.
bSI	buildingSMART International	
class diagram		A class diagram in the Unified Modelling Language is a static structure diagram that describes the structure of a system by showing the system's classes, their attributes and the relationships among objects.
Concept template		Part of MVD, defining how recourses should be linked to entities, restricting use of spatial structure, and other examples. Often described as part of an mvdXML file.
EA	Enterprise Architect	Software tool used to create and share definitions. In buildingSMART used to create Class diagrams of IFC.
Exchange Requirements		In this document: the selection of entities, properties, and other data to create a minimum set that Software Vendors need to support if they want to apply for Software Certification. Can be part of an mvdXML file, but often shared in pdf, excel, or other formats.
GitHub		Online service from Microsoft to collaborate on projects.
IDM	Information Delivery Manual	ISO 29481-1; a data standard independent methodology to describe business processes, use-cases and information requirements.
IDS	Information Delivery Specification	An XML based standard to define IFC based information requirements and how they should be delivered using IFC.
IFC 4	Actually 4.0.2.1, also known as IFC4 ADD2 TC1	Every reference to IFC 4 in this document is to IFC version 4.0.2.1, better known as IFC4 ADD2 TC1.
ISO	International Organisation for Standardisation	ISO is an independent, non-governmental international organization with a membership of 165 national standards bodies.
MVD	Model View Definition	A subset from IFC, plus additional requirements (concept templates) and a definition of a conformance level for software.
Pset		A standardized set of Properties, defined by buildingSMART and part of the IFC standard.
SCTE	Standards Committee Technical Executive	the highest technical policy and advisory board of buildingSMART International