Guideline

Date: 2021-06-28

Statsbygg - Postboks 232 Sentrum, 0103 Oslo, Norge
www.statsbygg.no/bim - bim@statsbygg.no
# Contents

1. **Introduction** .............................................................................................................................. 3
   1.1 Background and Purpose ............................................................................................................. 3
   1.2 SIMBA 1.3.1 Structure .............................................................................................................. 3
2. **SIMBA 2.0, part A – Machine Readable Requirements** ............................................................ 3
   2.1 About the Requirement database ........................................................................................ 3
3. **SIMBA 2.0, part B – General requirements in SIMBA 2.0** ......................................................... 6
4. **SIMBA 2.0, part C – Guidance to requirements in SIMBA 2.0** ................................................... 7
   4.1 A-1 Clarification of requirements for model practice/practices ............................................. 7
   4.2 A-2 Interdisciplinary Tag System (TFM) ................................................................................ 8
   4.3 A-3 Process Status Code .................................................................................................... 8
   4.4 A-4 Global Trade Item Number (GTIN) ................................................................................. 9
   4.5 A-5 Requirements for Landscape models .............................................................................. 10
   4.6 A-6 Triggers ...................................................................................................................... 12
   4.7 A-7 Requirements for principal / core disciplines ................................................................ 15
   4.8 The use of spatial zone objects .......................................................................................... 17
   4.9 B-1 Machine validation ...................................................................................................... 18
   4.10 B-2 Model delivery to archive ........................................................................................ 19
   4.11 B-3 As-Built ..................................................................................................................... 20
   4.12 B-4 IFC4 model deliverables ............................................................................................ 20
   A. Annex A – Interdisciplinary Tag System ................................................................................. 22
      A.1 Version of TFM system ...................................................................................................... 22
      A.2 TFM Property Set ............................................................................................................. 22
      A.3 TFM Properties ............................................................................................................... 22
   B. Appendix B – Model development process .............................................................................. 26
   C. Appendix C – Details related to requirements for principal / core disciplines ....................... 31
1. Introduction

1.1 Background and Purpose

With SIMBA 2.0, Statsbygg introduces the first version of requirement specification and validation of BIM deliverables for IFC4. The set of requirements also includes a new methodology for communicating the need for information on given objects between the disciplines (Trigger properties see section 4.8 in this document).

The SIMBA 2.0 requirements are based on Statsbygg’s BIM Manual 1.3 released January 1st, 2020. A series of additional requirements are included for the purpose of better utilizing BIM information throughout project life cycles.

1.2 SIMBA 1.3.1 Structure

SIMBA 2.0 consists of three main parts:

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Machine readable requirements</td>
<td>Requirement database</td>
</tr>
<tr>
<td>B</td>
<td>Non-machine-readable requirements</td>
<td>A list of general requirements provided in an Excel spreadsheet.</td>
</tr>
<tr>
<td>C</td>
<td>Guidance to requirements</td>
<td>This document.</td>
</tr>
</tbody>
</table>

Requirements in SIMBA part A and B are mandatory and are applicable unless otherwise is agreed specifically. In case of conflict between requirements, part A oversteers part B.

The requirements in SIMBA 2.0 part A and part B are SHALL requirements and must be followed in all projects unless otherwise agreed. The requirements are structured by domain and project stage. More information about Statsbygg's project model can be found here: https://statsbygg.metierportal.no/prosjektmodell/

In case of conflicting requirements, the requirements in Part A take precedence over the requirements in Part B.

2. SIMBA 2.0, part A – Machine Readable Requirements

2.1 About the Requirement database

Part (a) is a database holding requirements for object classes and their properties, structured after the IFC4 Schema. The database is available online via https://statsbygg-bim-q.com. Access credentials are required.

The database specifies requirements for each domain and project stage. Export of requirement sets can be saved to the open source format mvdXML or to human readable .pdf or .odt.
The database specifies generic requirement sets. Project can adapt, within limits of overall SIMBA principles, the requirements to their need. If no other requirements are agreed, the generic requirements apply. SIMBA 2.0 requirements are provided here [Norwegian version only]: https://sites.google.com/view/simba-bim-krav, look under the SIMBA 2.0 menu.

The project’s IFC model files shall undergo machine validation according to the mvdXML requirement set for the applicable project stage. The files must pass this validation before delivery. Any deviation from requirements in SIMBA 2.0 must be approved by Statsbygg. Early start of and continuous validation and correction of BIM models are recommended to avoid accumulation of errors towards the end of project phase.

mvdXML requirement sets are made available in all projects.

Descriptions on practical use of the database and validation can be found at: https://sites.google.com/view/simba-bim-krav [Norwegian version only]

### Table 2 – Machine readable requirements in SIMBA 2.0

New requirements refer to requirements in addition to Statsbygg’s BIM Manual 1.2.1. The guidance only provides guidance to these requirements. Requirements in Statsbygg’s BIM manual 1.2.1 is considered well established practice in the industry and should not require further guidance.

<table>
<thead>
<tr>
<th>Clause</th>
<th>Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-1</td>
<td>Modell practice</td>
<td>Clarification of model practice.</td>
</tr>
<tr>
<td>A-2</td>
<td>Interdisciplinary Tag System (“TFM – Tverrfaglig merkesystem”)</td>
<td>All entities shall be identified according to Statsbygg’s Interdisciplinary Tag System. Also see Annex A.</td>
</tr>
<tr>
<td>A-3</td>
<td>Process status coding</td>
<td>All entities shall be coded with process status. Also see annex B.</td>
</tr>
<tr>
<td>A-4</td>
<td>Product type coding - Global Trade Item Number (GTIN)</td>
<td>Each project shall decide whether to fill in the GTIN code delivered as part of the product documentation.</td>
</tr>
<tr>
<td>A-5</td>
<td>Requirements for landscape model</td>
<td>Statsbygg’s requirement database includes a requirement set for the landscape domain.</td>
</tr>
<tr>
<td>A-6</td>
<td>Triggers</td>
<td>Methodology and requirement sets to communicate the need for input from other domains and optionally additional requirements.</td>
</tr>
<tr>
<td>A-7</td>
<td>Requirements for conditional domains</td>
<td>Statsbygg’s requirement database includes requirement sets for the fire safety and acoustical domains.</td>
</tr>
</tbody>
</table>

1 «New» requirements refer to requirements in addition to Statsbygg’s BIM Manual 1.2.1. The guidance only provides guidance to these requirements. Requirements in Statsbygg’s BIM manual 1.2.1 is considered well established practice in the industry and should not require further guidance.
| A-8 | Zone objects | Requirements to the use of zone objects to report **net and gross area**. |
3. SIMBA 2.0, part B – General requirements in SIMBA 2.0

General requirements, which are not machine readable, are listed in a separate spreadsheet. In addition to these, the following new non-machine validating requirements apply to deliveries. These are described in the guide in part 4.

Table 3 – Non-machine-readable BIM requirements in SIMBA 2.0

<table>
<thead>
<tr>
<th>Ref.#</th>
<th>Theme</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-1</td>
<td>Machine validation</td>
<td>In all projects that use validation of machine-readable requirements on mvdXML format, the IFC deliverables shall undergo validation against requirements and errors are to be corrected, before agreed milestones</td>
</tr>
<tr>
<td>B-2</td>
<td>Model delivery to archive</td>
<td>Complete and quality-controlled BIM models shall be delivered according to current procedure for archiving in Statsbygg at the end of each project phase.</td>
</tr>
<tr>
<td>B-3</td>
<td>As-built model delivery</td>
<td>As-built model, corrected for all building changes compared to the approved detail design model, shall be delivered at project completion.</td>
</tr>
<tr>
<td>B-4</td>
<td>Delivery of IFC4</td>
<td>Models must be exchanged and delivered on IFC4 format with buildingSMART International certified Reference View (RV) export software.</td>
</tr>
</tbody>
</table>
4. SIMBA 2.0, part C– Guidance to requirements in SIMBA 2.0

4.1 A-1 Clarification of requirements for model practice/practices

The following is a clarification of model practice/practices for SIMBA 2.0.

(a) All relevant object classes shall be exported to IFC with «Base Quantities» (quantities such as lengths, widths, heights, areas, volumes, etc.). This requirement is a general requirement that is not validated.

(b) SIMBA 2.0 sets requirements for the use of a model for energy analysis as part of the design process. This requirement entails that in SIMBA 2.0 requirements are set for u-value (“Thermal transmittance”) on object types that are part of the climate shell (applies to the object classes decks, walls, windows, doors, roofs, columns, beams, curtain walls).

(c) Railings and handrails (IfcRailing) are interpreted as being complementary parts of walls, parapet walls, stairs and are included in the validation.

(d) When modeling ceilings, both cladding (IfcCovering) and cover (IfcSlab, PredefinedType = CEILING) can be used as object class. IfcCovering is also used to model other types of surfaces (floor coverings, wall cladding insulation, membranes, etc.). The same requirements apply to IfcSlab and IfcCovering.

(e) Openings (recesses, etc.) are assumed to be created automatically as IfcOpeningElement by the modeling software when e.g. a door object (IfcDoor) or window object (IfcWindow) is established in a modeled wall (IfcWall). It is not validated whether windows and doors are in an opening in the wall they are placed in.

(f) When modeling a recess (IfcOpeningElement) to report a need for a recess, the opening object must be named so that this is clearly understood, e.g. «Desired recess», «void» or similar. How this is named is to be agreed in the project. IfcOpenings are not validated.

(g) Descriptive name “Description”. User defined description of element type, its material and when applicable composite. Description shall communicate all properties relevant for cost and construction that are not communicated by other object properties. Applies to disciplines Architect (ARK), Structural Engineer (RIB) and Landscape Architect (LARK): If the software does not allow usage of Description, Statsbygg accepts usage of IfcMaterial for describing the object type. Usage of Description or Material shall be consistent for all objects and all models from the same discipline. Applies to disciplines Mechanical and Plumbing Engineer and Electrical Engineer: To the extent the Name sufficiently communicates all properties relevant for cost and construction that are not communicated by other object properties, the Description requirement can be omitted. This shall be confirmed by the appointing party.

(h) If a curtainwall (IfcCurtainwall) is modeled as a wall (IfcWall), typically in the early stages, the wall requirements apply to the curtainwall objects.

(i) HVAC must model outdoor routes in the ground (culverts). These must be modeled as spaces (IfcSpace.InteriorOrExteriorSpace: External). The HVACK model is not validated for space objects.

(j) If product-specific solutions are used, the product name / type must be stated in the tag attribute of the IFC model, e.g. IfcDuctSegmentType.Tag. Object type name (IfcRoot.Name) must still be product generic.
Some software uses the Tag attribute as the default attribute for other purposes, so the use of the Tag attribute should be tested, and quality assured/controlled, in the project. (k) Models used as a basis for procuring of contractor and subcontractors must be product generic. This means that you can not specify specific products in the model.

4.2 A-2 Interdisciplinary Tag System (TFM)

4.2.1 Use of TFM in BIM

TFM is a tag system developed by Statsbygg to describe technical systems and manage links between physical products in a building with documentation. The use of TFM is increasing and a requirement from several other Norwegian clients and owners.

Population of TFM in BIM systems and objects opens the possibility to i.e. visualize system affiliation and access documentation from BIM models.

There are currently different interpretations of TFM tags, dependent on the operating organization receiving BIM models after finalization of a project. Prevailing version for projects in Statsbygg is “PA 0802 TVERRFAGLIG MERKESYSTEM (TFM)”, version from 2017. Statsbygg will modify the TFM requirements in accordance to the upcoming official Norwegian standard for TFM (NS-TFM), expected to be released in 2020.

If the project decides to use Statsbygg’s PA 0802, see Appendix A, point 5.3.5 “Use of PA 0802 in model).

NS-TFM is more precise with respect to methodology and type specification. It is also adapted for use with BIM models and requirement specification tools. It is recommended to use NS-TFM for all new projects implementing a TFM standard. Projects must decide which TFM version to use with respect to requirements for documentation for Management, Operation, Maintenance and Development (NO: FDVU-dokumentasjon).

If the official version of NS-TFM is not available, the preliminary hearing version can be used. It will be made available on request to the BIM group, BIM@statsbygg.no.

TFM properties described in Appendix A are pursuant to the preliminary hearing version of NS-TFM and are used for projects using NS-TFM.

For more information about Interdisciplinary Tag System (TFM) used with BIM, refer Appendix A.

4.3 A-3 Process Status Code

The maturity of a BIM object, with respect to the decision and quality assurance process, is not visible by its geometry alone. Objects are often imported from software libraries and will rarely change the level of geometric detailing in the course of a project’s phases. An object can have the same level of geometric detailing although it has changed dimension, location, type, material etc. throughout the course of a project.

BIM is becoming a more integrated part of project processes, in the construction phase as well as Management, Operation, Maintenance and Development. Continuous communication of the maturity of an object is therefore important. Can the object be used as a basis for cost estimates or purchase? Can it be built as modelled?
A status system is used for precise communication of an objects maturity. The status is held by an object property. Various status systems for model maturity exists. The contractors’ association “Entreprenørforeningen Bygg og Anlegg” (EBA) issued in October 2018 the first version of the MMI guide “MMI-veileder” together with the consulting engineers’ association “Rådgivende Ingeniørers Forening” (RIF) and the architects’ association “Arkitektbedriftene”. Statsbygg requires the use of Model Maturity Index (MMI) as default unless other is agreed.

For more information on model maturity, MMI codes and Level of Geometry (LoG), refer Appendix B.

4.4 A-4 Global Trade Item Number (GTIN)

Global Trade Item Number (GTIN) is an identification code managed by the organization GS1. It is used to give commercial product types a unique number, rendering the confusion of types of products impossible when using GTIN. The code can also be used to track products documentation. This reduces the risk of errors with orders, sales and other types of product handling in the value chain.

Statsbygg wants to locate product documentation using BIM objects. It is possible to connect BIM objects to commercial products, either by assigning GTIN codes directly to a model property or by including GTIN codes as product documentation found using BIM objects TFM code.

GTIN is a unique identification code for product types. It is also possible to identify a single occurrence of a product by a SGTIN code (Serialized GTIN). SGTIN is not a standard requirement if not agreed upon in the project. For more information on GTIN see www.gs1.no.
4.5 A-5 Requirements for Landscape models

Requirements for Landscape models and model for outdoors in SIMBA 2.0 are based on previous standardization work carried out in Norway for Landscape Architecture, such as «BIM for Landscape Architecture» and «SOSI Landscape Architecture».

In these standardizations, most of the main objects that Landscape Architects use to model a landscape model are described and defined. The standards also contain pre-sales for properties associated with the objects, and when these properties should be defined in the project based on Statsbygg’s design phases. Each of the main object types is described with separate PDTs, Product Data Templates, which can be downloaded from the website, http://bimforlandskap.no/.

Similarities between Architecture and Landscape Architecture have been looked at throughout the standardization period. This is well described in "IFC for LARK - Example used in Revit", http://underland.no/?page_id=730 from 2012.

By "reusing" object types in IFC format from Architecture, Landscape Architecture can carry out a faster standardization process. Projects where outdoor and landscape architecture are integrated with buildings and structures have shown that they can go a long way in exchanging landscape models in IFC format simply by using existing objects and object types.

The first proposal for how previously defined objects for Landscape Architecture could be exported via existing objects and object types in IFC was presented in 2018 on the website http://bimforlandskap.no/ifc-for-landskap/.

Figure 3 – The pictures show excerpts from previous standardization work within BIM for Landscape Architecture. The example shows how a tree has different requirements for geometry and properties in different design phases.
In IFC, many object types have a more detailed division into what the object type is. For example, IfcFurniture can be divided into subgroups with IfcFurnitureTypeEnum: CHAIR, TABLE, DESK, BED, FILECABINET, SHELF, SOFA, USERDEFINED and NOTDEFINED

It is important to point out that when Landscape Architects reuse IFC main types such as IfcFurniture, own types can be linked to the object. The method is described in a separate appendix. Many of the landscape objects are described in SIMBA 2.0 as custom object types related to a more general overriding object type. The set of requirements for Landscape Architecture in SIMBA 2.0 is Statsbygg’s proposal for a common method for reusing existing IFC4 Add2.

Figure 4 – Display of the SIMBA 2.0 requirements database as presented in BIMQ. The vegetation object is displayed with properties, code and description. Furthermore, it is shown which IFC entity is used together with a custom object type that Statsbygg wishes to use for SIMBA 2.0 IFC4 Add2 and in which design phase the properties of the object must be presented in the model.

SIMBA 2.0 has taken it a step further by defining objects, object types and properties based on what are described as separate items for description, pricing and management in NS3420-K: 2019 Landscaping and NS3420-ZK: 2019 Management and operation of parks and landscape areas. Many of these object types are not required in SIMBA 2.0.

The goal is to promote a common Norwegian way of defining Outdoor and Landscape models in openBIM. A number of companies have worked for a common Norwegian approach for many years and Statsbygg wants to contribute with SIMBA 2.0. The solution presented in SIMBA 2.0 is based on IFC4 Add2. The proposed naming of objects is adapted to this standard. IFC4.3 RC1 dated 2020-04 contains several object types specifically for Landscape Architecture. Some examples are IfcPlant, IfcPavement, IfcKerb, .. Naming of custom types for Landscape Model in SIMBA 2.0 is adapted to IFC4.3. 
https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/
4.6 A-6 Triggers

The term "triggers" has been introduced in the SIMBA 2.0 requirements sets. In the BIMQ database, triggers appear in some of the NOSSB property sets.

The following is a more detailed explanation of what triggers are and how they are intended to be used.

4.6.1 The NOSSB_ReqTriggers property set and associated properties

In SIMBA 2.0, the custom property set NOSSB_ReqTriggers exists for a variety of object types. The prefix «NOSSB» follows the requirement in NS 8360 that custom property sets must have up to five characters. We have chosen to use the first two characters to indicate that this is a Norwegian property set, where «NO» follows the ISO 3166-1 alpha-2 country codes. «SSB» can be interpreted in several ways, but the development of the requirements set is a collaboration between Sykehusbygg and Statsbygg. BaneNor is considering following the same set of requirements. «Req» refers to "Requirement", ie a requirement.

In the BIMQ database, the example looks like this for a pump – IfcPump:
As shown, a set of properties that say «Has» is made available under this property set. In the example with the pump, there is a tick for the requirement «Has controls connection» and «Has electrical connection», or some of the delivery points. This means that the HVAC designer of the relevant model in which the pump is included must consider whether the pump should be connected to an SD system («controls») or not, and whether it has requirements for electrical connection or not («electrical»). The job of the designer to put «YES» on all pump objects (IfcPump) to be connected to the SD system (probably not all, but quite a few?), And «NO» on the rest. For electrical connection it will be a similar task (probably all, unless the pump is part of a larger assembled object that gets electrical connection via another object type).

The purpose of only deciding on «should have” or «do not need” some connection or supply is that the models from the various disciplines can be used for effective interdisciplinary communication as well as «counting” needs relatively early in the design. In the example, the HVAC designer can tell the electrical designer that "I want these pumps in the SD system", without having to take a detailed position on exactly how the connection should be solved with communication protocols and interfaces. This is thus the "basic" trigger function that is intended to be used, especially in the earlier design phases.

4.6.2 Other property sets «which are triggered»

The second trigger function is that when a trigger property, as indicated above, is set to «YES» / «TRUE», e.g. that NOSSB_ReqTriggers.HasControlsConnection=TRUE it will "trigger" that another related / relevant - property set should be included in the validation. In the present example, this may be e.g. NOSSB_BuildingControls, which contains a number of more detailed automation features.
Within the property set, one can thus set a property requirement at a given milestone / phase or take one step further and validate the permitted values for the property in question, as shown in the example above. It is not mandatory to «tick» of requirements, but for the objects where NOSSB_ReqTriggers.HasControlsConnection=TRUE is set, the properties in NOSSB_BuildingControls that are «ticked» (required) for a delivery will be validated. Without ticks in NOSSB_BuildingControls no further requirements will be triggered.

The typically intended use of «triggers» is thus the following (based on the example above):

1. In the requirements set for HVAC, it is ticked off that in an early design phase NOSSB_ReqTriggers.HasControlsConnection is to be set for pumps, that is, the HVAC Designer must set either TRUE or FALSE to this property for all IfcPump objects to avoid validation deviations. This gives the HVAC Designer the opportunity to tell the Electrical Designer that «I want these pumps in the SD system».

2. Later in the design process when you have chosen relevant solutions to the need, and typically made system choices for what kind of SD system to have, it can be required that the HVAC designer must specify which of the pumps, previously defined to have SD connection, should have BACnet protocol and which should have LON protocol. The selection is made in NOSSB_BuildingControls.BUSProtocolType, ie the requirement for OK validation is that «this field must have a value». By writing «Merry Christmas» in the field, it will be OK in terms of validation, as long as there are no requirements for valid values.

3. Requirements can be detailed by listing the allowed values in the requirements set, so that NOSSB_BuildingControls.BUSProtocolType=BACnet will return OK in the validation, while NOSSB_BuildingControls.BUSProtocolType=Merry Christmas will give validation deviations.
4.7 A-7 Requirements for principal / core disciplines

4.7.1 Definition of principal / core disciplines
In SIMBA, we define premise discipline as follows:

- Principal / core disciplines - are disciplines that specify overall requirements / concepts / premises that are the basis for other disciplines’ detailed design.

In a BIM perspective, principal / core disciplines will be limited to disciplines that can provide information to a model, but who do not own physically constructable objects in a model.

Typical premise subjects are acoustics, fire safety, but can also include e.g. energy, safety ("security" - perimeter security, explosion safety, etc.) and environment (BREEAM, greenhouse gas). Other topics that may also fall into this category include requirements / information in connection with logistics and waste management.

4.7.2 General principles for requirements from principal / core disciplines
Requirements from principal / core disciplines are entered into the BIM model by filling in various parameters in the form of text or Boolean variable (TRUE / FALSE). In the database (BIMQ), parameters have been proposed that may be relevant to enter in the BIM model for Acoustics and Fire Safety. The list of parameters is a larger selection than what is natural to include in a standard project and should facilitate individual projects where it is practical to include the selected parameters, possibly a future increase in the desired amount of information in the BIM model.

The parameters to be filled in will vary from project to project and will depend on the design phase. The parameters to be included are decided by the project manager in consultation with the project’s BIM manager from the relevant premise discipline. The standard completed system has an absolute minimum of requirements that must be included in a project in order for it to be valuable for the premise discipline to have its own model.

For a premise discipline model, there are mainly three different categories of «BIM objects» that may be relevant to enrich with parameters: physically constructible objects, space objects and virtual objects.

*Physically constructible Objects (PO)*
POs are objects as they appear in ARK / main discipline models. Relevant objects are, for example, walls, doors, windows and covers.

*Space Objects (SO)*
SOs are objects that constitute a volume limited by surrounding building parts that define the use of an area as it appears from the model. In this context, it is first and foremost space objects (IfcSpace) in the architect model that are relevant.

*Virtual objects (VO)*
These are objects that do not usually appear from the architect / Master discipline model and that are not built. Typical variants are:

- Objects that make up a custom volume. Can be similar room objects, collection of rooms or a volume that suits a specific requirement.
- Objects that act as duplicates of architect / main discipline objects.
Objects generated by interfaces to a volume. In an IFC model, these are in practice IfcVirtualElement generated by the IfcRelSpaceBoundary relationship type.

4.7.3 Method for filing requirements from the premise discipline fire safety
Entering requirements from the fire safety designer takes place by objects in the fire technical model being enriched with information. Only VO and SO will be used according to the description under 4.9.2. To ensure that the database searches for information on the correct objects, requirements must be placed on defined IFC object types as set out in this guide.

In a fire technical model, there will mainly be four groups of parameters (requirements / information). These are:

1. NOSSB_FireSeparator
2. NOSSB_FireSectionAndCompartment
3. NOSSB_FireInformation
4. NOSSB_FireEscapeAndRescue

For a more detailed description of the groups and what kind of objects the requirements are placed on, see appendix C.1

4.7.4 Method for filing requirements from premise discipline acoustics
Entering requirements from acoustics takes place as a combination of objects in their own acoustic model being enriched with information and objects in other disciplines' models being enriched with acoustic information. A maturation phase is underway in the acoustics discipline, and the practice varies between the consultants in the industry. The acoustics discipline also works with solutions in relation to the satisfaction of requirements, and thus different requirements depending on what and who is to use the information from the model. The combination has meant that the method is currently set up to be flexible and ensure the possibility of adding more requirements to each project. Only minimum requirements have been set for airborne sound insulation for interfaces in the pre-project phase and detailed project phase, related to requirements in TEK / NS 8175.

In an acoustic model, there will mainly be three groups of requirements / information.

1. NOSSB_AcousticSeparator
2. NOSSB_AcousticSectionAndCompartment
3. NOSSB_AcousticInformation

For a more detailed description of the groups and what kind of objects the requirements are placed on, see appendix C.2

It is desirable that projects include different types of parameters for acoustics in order to gain experience and be able to form a best practice. An example, related to airborne sound insulation requirements, may be that in the sketch project phase requirements can be set for airborne sound insulation at room level, and then set requirements for airborne sound insulation for interfaces in the pre-project / detailed project phase and possibly on specific walls / doors / windows in the detailed project or FDV information.
4.8 The use of spatial zone objects

4.8.1 About IfcSpatialZone

IFC4 has introduced a new instance entity (object type) named IfcSpatialZone.

It is described as follows in the IFC scheme:

«A spatial zone is a non-hierarchical and potentially overlapping decomposition of the project under some functional consideration. A spatial zone might be used to represent a thermal zone, a construction zone, a lighting zone, a usable area zone. A spatial zone might have its independent placement and shape representation.”

In other words, IfcSpatialZone has its own, independent geometry representation, in contrast to IfcZone which is only a grouping function (container) for IfcSpace (space objects) or other IfcZone (zone-in-zone), and thus "inherits" the geometry of the space objects included in the zone. IfcZone can thus not be used to represent geometry other than the sum of geometries for the space objects.

The advantage of using IfcSpatialZone is that it can be used to represent any volume that has one or another common denominator, for example a control zone with a delimitation in the middle of a wall, an air-conditioned zone that also goes a bit into a wall, a rental zone that f.ex. is based on area from the outer edge of the outer walls to the middle of the inner walls, a rot-damaged area on parts of a wall or which partially covers several walls and a slab, etc.

IFC4 also has a corresponding type object IfcSpatialZoneType, and this in turn has predefined types, expressed by an enumeration (pull-down) IfcSpatialZoneEnum. The predefined types are:

Among the predefined types are i.a. fire zones (FIRESAFETY) and safety zones (SECURITY). In addition, there is a USERDEFINED type, to establish custom zone types.

When, in an IFC model, you set IfcSpatialZoneEnum=USERDEFINED, the "IFC mechanism" is that at the same time a custom "name" is set in the ObjectType attribute of the current instance. In this case it will be IfcSpatialZone.ObjectType = <zone name>.

4.8.2 Requirements to use IfcSpatialZone to set gross floor area (GFA) and usable area (UA) per story

Statsbygg has currently chosen to define two types of custom zone types using the entity IfcSpatialZone (several types may come later):
IfcSpatialZoneTypeEnum=USERDEFINED with IfcSpatialZone.ObjectType=GFA is the requirement to set custom «zone name» GFA (Gross Floor Area) to represent the gross area BTA according to NS 3940. The requirement applies per defined floor object (IfcBuildingStorey) in the model. There must therefore be an IfcSpatialZone object for each floor that geometrically covers the entire floor, to the outer edge of the outer wall.

IfcSpatialZoneTypeEnum=USERDEFINED with IfcSpatialZone.ObjectType=UA is the requirement to set custom «zone name» UA (Usable Area) to represent the usable area BRA according to NS 3940. The requirement applies per defined floor object (IfcBuildingStorey) in the model. There must therefore be an IfcSpatialZone object for each floor that geometrically covers the entire floor, to the inner edge of the outer wall.

Statsbygg wants to use English naming of the custom zone types to ensure good dialogue also with foreign design groups and in foreign projects.

The criteria for defining a floor object (IfcBuildingStorey) in a model shall, unless otherwise agreed, follow the rules for floor number in «Guidance: Degree of utilization - Calculation and measurement rules» from the Ministry of Local Government and Modernization (KMD) from 2014.

The requirement to use IfcSpatialZone to express GFA and UA in IFC models replaces previous requirements to have space objects (IfcSpace) with names (.Name) respectively BTA and BRA.

4.8.3 Other uses of IfcSpatialZone
In each project, it is possible to define additional custom zone types for other types of zones that are relevant. Some suggestions / examples of current zone types ("name" for attribute ObjectType) may be:

- CLEANROOMLEVEL = zone with clean room requirements
- CULTURALHERITAGE = cultural heritage zone
- INFECTIONCONTROL = infection control zone
- LEANCONSTRUCTIONCONTROL = control zone for Lean Construction
- PRESSURECONTROL = sone med krav til lufttrykk (overtrykk/undertrykk)
- UNIVERSALDESIGN = Zone for degree of fulfillment for universal design

4.9 B-1 Machine validation
From 1st of January 2020 Statsbygg requires that all projects using BIM models shall perform machine validation and correction of errors with respect to the prevailing machine-readable requirements.
1. Requirements are formulated in a requirement database, describing requirement for specific object classes, phase and responsible discipline. When a new project is started, a project-specific copy is made in which project-specific adjustments can be made.

2. Designers and model makers can read requirements either directly in the requirements database or in reports. They make model according to requirements. The model is submitted for check on the open format IFC.

3. Requirements can be expressed as machine-readable requirements on the open format mvdXML. The mvdXML requirement set is imported to a validation tool for model check.

4. Model is checked with the validation tool.

5. Errors are reported and can be communicated on the open BIM collaboration format (BCF).

6. The model is ready for delivery when all non-acceptable errors are corrected (Provided that all non-machine-readable requirement also are satisfied). It is required that the projects themselves, alternatively with some training, carry out machine validation of models before deliveries.

Machine readable requirement sets will be made accessible in all projects on mvdXML format.

Any deviations from requirements in SIMBA 2.0 must be agreed upon with project or building client/owner.

To avoid accumulation of errors towards delivery deadline, it is recommended to perform continuous validation of models.

Use of the database and validation tool can be found at: https://sites.google.com/view/simba-bim-krav

4.10 B-2 Model delivery to archive

All modelling disciplines shall, after every project phase, deliver a complete set of quality assured models for archiving. Quality assurance according to requirements in SIMBA 2.0 and by use of machine validation B-1 is required. Models shall be delivered in accordance with the prevailing routine for archiving.
Delivery shall include metadata about project and models.

4.11 **B-3 As-Built**

It is, in SIMBA2.0, part B, required delivery of as-built models for all Statsbygg projects. Minimum requirements for as-built models are:

- Model corrected for all building changes from the approved detail design model to project completion. This applies for changed concepts, types, positions beyond accepted tolerances.
- Properties specified for as-built delivery in the requirement database is included. I.e. unique, quality assured TFM strings and updated process status codes (MMI) reflecting as-built maturity. Product type codes (GTIN) is included if agreed in the project.

An example of workflow for model development from as-designed to as-built is shown in the figure below. The workflow for reaching as-built status must be decided by the project. Contractors, designers and client form a joint work group responsible for registration of deviations and decided actions. The work group come to agreement on allowed deviations, methods for control and frequency of deviation management sessions.

Correspondence between the physical construction and model is continuously monitored as part of the contractor's quality assurance. Deviations are reported at given intervals to the joint work group. The work group decides on how to handle rejected deviations and forwards the decision to the contractor or designer. When all rejected deviations are handled and the model only consists of approved deviations, status can be changed to as built.

![Figure 8 - Example of workflow for BIM model from as-designed to as-built](image)

4.12 **B-4 IFC4 model deliverables**

SIMBA 2.0 sets requirements for model deliveries on IFC4 Add2. The requirement includes that the quality of all model deliveries must be in accordance with the specification for buildingSMART's Reference View for export. In practice, this means that software that is not certified for IFC4 Reference
View Export should not be used in the project unless it is assured that the quality of model exports is satisfactory. The supplier of the information is responsible for the quality of the model export.
A. Annex A – Interdisciplinary Tag System

A.1 Version of TFM system

If the project chose to deliver TFM coding according to Standards Norway’s standards NS 3457-7, -8 and -9 and NS 8360-2 with guidelines (in this document referred as NS-TFM), the minimum model requirement is Level 0 as described in clause A.3.1.

If the project chose to deliver TFM coding according to Statsbygg’s PA 0802 the minimum model requirements is Level 0 as described in clause A.3.5.

A.2 TFM Property Set

The Norwegian Interdisciplinary Tag System (TFM) is a classification, identification, and physical labelling system, widely used in the Norwegian AEC sector. The TFM system is developed upon ISO/IEC/CEN/CENELEC standards and has similarities with the ISO 81346 series standards. The actual system and component codes are however Norwegian.

Name of Property Set: NOSSB Reference

NO = Norway, SSB = Acronym for Norwegian clients Statsbygg + Sykehusbygg – according to rules for user defined properties in Norwegian Standard NS 8360, clause 5.3

All properties for TFM shall reside under this property set name.

A.3 TFM Properties

Three levels exist for adding a TFM identification in BIM.

- At Level 0 the complete (concatenated) TFM code string is specified as a property.
  - All preceding identifiers are included as part of the string.
- At Level 1 each «topic» (section) in the TFM code string (like placement/location, system, component occurrence, component type) is separated as properties.
  - Identifier prefix for each “topic” (section) is not included as part of the code but is allocated by software.
  - Identifiers between parts within each “topic” (section) are included as part of the code.
- At Level 2 all parts within each “topic” (section) in the TFM code string are included as separate properties.
  - All identifiers that separate the parts are not included as part of the code but are allocated by software.

A.3.1 Level 0

Level 0 is required for Statsbygg projects – based on needs identified from FM & operations. Even though Level 0 is the required deliverable it may be easier to develop and maintain the codes as separate data fields – either at Level 1 or Level 2. Each project should assess coding at Level 0, 1, or 2. The assessment may be done separately by discipline.

The advantages of coding at Level 2 could simplify:

- Development of sequential numbering schemes
• Extracting the parts that are relevant for physical labelling

If the parts are included as separate data fields they are "assembled" (concatenated) to a code string when delivered to the client. The following properties shall be used for specifying TFM codes at Level 0, 2, and 2 respectively.

Table A.1 - TFM, Level 0

<table>
<thead>
<tr>
<th>Level 0</th>
<th>Description</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total (aggregated, concatenated) code</td>
<td>RefString</td>
</tr>
</tbody>
</table>

Example of code at level 0:

- NS-TFM: +++B106=360.014.04-SQZ0023\%SQZ.008.03
- Or
- PA 0802 type +B106=360.014-SQ023
- PA 0802 occurrence +B106=360.014-SQ008T

A.3.2 Level 1

At Level 1 the individual «topics» (sections) of the TFM code are included; Placement/location, system occurrence, component type – as data fields for each «topic» (section).

Table A.2 - TFM, level 1

<table>
<thead>
<tr>
<th>Level 1</th>
<th>Description</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Placement/location - site and building</td>
<td>RefPriSysLoc</td>
</tr>
<tr>
<td></td>
<td>Total (aggregated) system code</td>
<td>RefPriSysOcc</td>
</tr>
<tr>
<td></td>
<td>Total (aggregated) component occurrence ID</td>
<td>RefCompOcc</td>
</tr>
<tr>
<td></td>
<td>Total (aggregated) component type code</td>
<td>RefCompType</td>
</tr>
</tbody>
</table>

Example of code at level 1:

- NS-TFM ++ B106 = 360.014.04 - SQZ0023 % SQZ.008.03
- Or
- PA 0802 type + B106 = 360.014 - SQ023
- PA 0802 occurrence + B106 = 360.014 - SQ008 T
A.3.3 Level 2
At Level 2 all parts of the TFM code are included as separate data fields. Please remark that for the placement/location part the code is identical at Level 1 and 2 as it consists only of a client specific code.

Table A.3 - TFM, Level 2

<table>
<thead>
<tr>
<th>Description</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Placement/location – site and building</td>
<td>RefPriSysLoc</td>
</tr>
<tr>
<td>System code</td>
<td>RefPriSysClass</td>
</tr>
<tr>
<td>System type code</td>
<td>RefPriSysNo1</td>
</tr>
<tr>
<td>Sub-System code</td>
<td>RefPriSysNo2</td>
</tr>
<tr>
<td>Component code</td>
<td>RefCompClass</td>
</tr>
<tr>
<td>Component occurrence code</td>
<td>RefCompOccNo</td>
</tr>
<tr>
<td>Component type code</td>
<td>RefCompTypeNo1</td>
</tr>
<tr>
<td>Sub-Component type code</td>
<td>RefCompTypeNo2</td>
</tr>
</tbody>
</table>

Example of code at level 2:

++ B106 = 360 . 014 . 04 - SQZ 0023 % SQZ . 008 . 03

Or

+ B106 = 360 . 014 - SQ 023

+ B106 = 360 . 014 - SQ - SQ 008 T

A.3.4 Additional Codes
The use of additional codes is not required in general. The use of these codes should be agreed on in each project based on project needs.

Additional Codes are not part of the traditional, hierarchical TFM string, but may provide value to specify information that is important for FM & operations – e.g.:

Table A.4 - TFM, additional codes

<table>
<thead>
<tr>
<th>Description</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification code from Norwegian Standard NS 3420.</td>
<td>RefClassNS3420</td>
</tr>
<tr>
<td>Classification code from Norwegian Standard NS 3451.</td>
<td>RefClassNS3451</td>
</tr>
<tr>
<td>Door number according to the project’s system.</td>
<td>RefDoorNo</td>
</tr>
</tbody>
</table>
Unique identification (generated number in modeling tool or database) to synchronize component type between BIM and type in databases.

RefCompTypeDbSync

Unique identification (generated number in modeling tool or database) to synchronize component occurrence between BIM and type in databases.

RefCompOccDbSync

Complex - Owner's overall ID - Top node identifier.

Top node identifier should be used to distinguish between different complexes (composition of buildings or structures in one location or property) in the owner's total data register, so that the owner within the individual complex is free to use codes for buildings, wings etc. without taking into account the codes used in other complexes.

RefComplex

System component number.

RefPriSysComp

Identifier communicating that this component occurrence is a system component that defines / starts a system.

RefSysCompOcc

Functional system (if relevant). The total functional system identification.

RefFunSys

Secondary systems (if relevant).

RefSecSysOcc

Technical location of the component. Unique identification of the component (occurrence) or system (occurrence) on which the component is located.

RefCompLocTech

Component location (space/spatial function). Space number.

RefCompLocRoom

Building number in accordance with Statsbygg's system.

RefCompLocBFac

Horizontal breakdown. Typical wing or section in accordance with defined breakdown structure.

RefCompLocHor

Vertical breakdown. Typical building storeys.

RefCompLocVer

Location (space) that the component serves/controls/operates. Space number.

RefCompLocServe

Local role or task within a system or a composite component.

RefCompRole

Standard system or system type.

RefSysType

Standard component.

RefCompStd

### A.3.5 Use of PA 0802 in model

If the project chose to deliver TFM coding according to PA 0802 Level 0 applies, where all TFM data fields are aggregated in one string, with the use of the following additional codes:

<table>
<thead>
<tr>
<th>Description</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total (aggregated, concatenated) string for component occurrence.</td>
<td>RefStringHist1</td>
</tr>
<tr>
<td>Total (aggregated, concatenated) string for component type.</td>
<td>RefStringHist2</td>
</tr>
</tbody>
</table>

### Table A.6 - Use of PA 0802 in model – Component code

| Additional Codes |
|------------------|----------|
| APPROVAL DATE    | 28.06.2021 |
| APPROVED BY      | Anders Fylling |
| PROFESSION AND METHOD |
| RESPONSIBLE UNIT  | FPROS |
| REVISION NUMBER  | 1 |

<table>
<thead>
<tr>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Andets Fylling</td>
</tr>
<tr>
<td>direktar faglig ressurssenter</td>
</tr>
</tbody>
</table>
B. Appendix B – Model development process

B.1 Model development process

Two pivotal processes during the design stage are model coordination and model control. This annex provides brief guidance to how it may be used to communicate the maturity of the information deliverable.

B.1.1 Model coordination

Model coordination is an iterative process where design concepts are modelled and tested against other disciplines and project constraints such as cost, energy balance etc.

Models are aggregated at an agreed frequency for geometrical coordination and other analysis. Geometrical clashes and other findings are rectified in the next iteration of mode design. While the models are developed the object’s process status will remain ‘In progress’.

B.1.2 Model control

Before a milestone delivery the quality of the information shall be ensured and possibly documented. The process changes from coordination to control. The model aggregation and checking may be the same. But now the design shall not be further developed except for rectifying geometrical clashes and other deviations from the requirements. The model objects can be process status coded for ‘Ready for
model control’ before the control to communicate when a group of objects or a part of the construction is ready for the control process. When the model control is executed, and all information are according to the requirements process status can get ‘Approved multi-disciplinary control’ communication that it is ready for the next stage.

Figure B.2 - Example on model control. The control is executed according to a control plan and documented with signed check lists. Objects are set per control area (agreed section of model, eg floor, department, system, etc.) to communicate maturity level.

B.2 Process status coding within BIM model
B.2.1 General
Statsbygg and Sykehusbygg has developed common terminology, property sets and properties for use of process status coding with BIM. The property set grouping all properties relevant for process status coding is NOSSB_Process (According to rules for establishing new properties in the Norwegian BIM standard NS 8360.

Available process status codes are DesignedStatus, ConstructedStatus and OperationalStatus.

The reason for splitting codes for design, construction and operation is that an object can have different status for the three perspectives in parallel. E.g. a product can be procured while still not have approved multidisciplinary control. That means that the location of the product can still be undetermined, but the properties and geometry is locked.

It is important that use of process status codes is unambiguously agreed according to the project’s needs and communicated in the project.

The process status code is applied to the individual object, but codes can be assigned to several objects simultaneously within a controlled area.

B.2.1 Model maturity index (MMI)
Statsbygg projects use the contractors trade Association EBA’s MMI guidance and process status codes as a default. If another code system is used. The process status codes shall use the same properties under NOSSB_Process with BIM.

The following table is a proposal for a further detailing of EBA’s MMI code system. The project decides which codes that shall be used.
Table B.1 - Proposal for further detailing of EBA’s MMI Codes.

<table>
<thead>
<tr>
<th>MMI Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMI100</td>
<td>Established</td>
</tr>
<tr>
<td>MMI200</td>
<td>Concept</td>
</tr>
<tr>
<td>MMI300</td>
<td>Ready for multi-disciplinary control - Geometry</td>
</tr>
<tr>
<td>MMI320</td>
<td>Geometry locked</td>
</tr>
<tr>
<td>MMI340</td>
<td>Ready for multi-disciplinary control – Geometry and information</td>
</tr>
<tr>
<td>MMI350</td>
<td>Multi-disciplinary control approved</td>
</tr>
<tr>
<td>MMI380</td>
<td>Final design</td>
</tr>
<tr>
<td>MMI400</td>
<td>Approved for tendering Production design*</td>
</tr>
<tr>
<td>MMI420</td>
<td>Product procured</td>
</tr>
<tr>
<td>MMI440</td>
<td>Ready for control for production</td>
</tr>
<tr>
<td>MMI450</td>
<td>Approved for construction Production design*</td>
</tr>
<tr>
<td>MMI460</td>
<td>Built and ready for as-built control</td>
</tr>
<tr>
<td>MMI470</td>
<td>Approved As-built control</td>
</tr>
<tr>
<td>MMI480</td>
<td>Ready for handover to client</td>
</tr>
<tr>
<td>MMI500</td>
<td>Approved handover to client</td>
</tr>
<tr>
<td>MMI520</td>
<td>Handover operation approved</td>
</tr>
</tbody>
</table>

Codes in bold are standard MMI Codes. Codes in italic are Statsbygg’s additions.

* In MMI guidance from EBA MMI400 means that the object is “Approved for construction”. Statsbygg has identified a need for at least two codes before “Approved for construction”: “Approved for tendering” and “Product procured”

A distinction is made between whether the model is interdisciplinary controlled only for geometry and for geometry and information. When geometry is controlled and locked, it is known that the project is geometrically coordinated and that it will be possible to build with specified object types without problems with geometric collisions between building elements on the construction site. Checking geometry presupposes that the geometry of the object types does not change. It may take longer to get all parameters in place e.g. coding, properties etc. When the geometry is checked and approved, the model can be communicated to e.g. contractor / supplier for production. The designers can subsequently work on supplementing information on objects.

Some of the status codes describe that the object is ready for validation. These communicates, from the person responsible for the object, to the person who is to carry out the validation, that the object is ready for control. Changes to an object after model validation can render the status code obsolete. All subsequent changes after a control shall therefor be agreed with the person responsible for the control. Codes describing that the object is ready for control are typically used in projects of a certain complexity.
and where the person responsible for the validation is not the same as the person responsible for the object.

The process status code is applied to properties in the NOSSB_Process property set according to the table below. The table specify the designated use of properties in the property set NOSSB_Process. As a default:

- The design team is responsible for all DesignedStatus properties;
- Contractors are responsible for ConstructedStatus properties;
- Construction operator is responsible for OperationalStatus properties.

Other use shall be agreed in the project.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DesignedStatus</td>
<td>Communicates the maturity of the object in the decision and quality control processes in the design phase. MMI Codes from MMI000 to MMI399.</td>
</tr>
<tr>
<td>ConstructedStatus</td>
<td>Communicates the maturity of the object in the decision and quality control processes in the construction phase. Codes from MMI400 to MMI499.</td>
</tr>
<tr>
<td>OperationalStatus</td>
<td>Communicates the maturity of the object in the decision and quality control processes in the operational phase. Codes from MMI500 and up.</td>
</tr>
</tbody>
</table>

**B.3 Object geometry**

**B.3.1 Level of geometry (LoG)**

SIMBA distinguish between requirements to geometry and to information. In SIMBA-s requirement database, BIMQ requirements to geometry are specified as Level of Geometry, abbreviated LoG. There are five levels of geometry from the start of the modelling process to as-built design. The requirement to geometry in BIMQ is not part of the machine validation of the models. The project shall establish a methodology to secure that the model geometry is in accordance with the LoG requirements.

The levels are as default similar for all elements. But requirements to when a new level is required may differ for the individual object classes. The following levels of geometry as default.
Table B.3 - Levels of geometry for elements

<table>
<thead>
<tr>
<th>LoG level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LoG Level 1</td>
<td>The object can be used for, but not limited to, e.g. visualization and assessment of outline conceptual design. The object represents either a conceptual geometry or a placeholder volume. The object geometry is a draft regardless of level of detail. Object location and orientation is conceptual.</td>
</tr>
<tr>
<td>LoG Level 2</td>
<td>The object can be used for, but not limited to, e.g. visualization and assessment of multi-disciplinary conceptual design. The object represents a generic type. Geometry, location and orientation is approximate.</td>
</tr>
<tr>
<td>LoG Level 3</td>
<td>The object can be used for, but not limited to, e.g. cost estimation and model coordination. The object represents both its types and instance. Geometry, location and orientation is exact after design coordination.</td>
</tr>
<tr>
<td>LoG Level 4</td>
<td>The object can be used for, but not limited to, e.g. procurement, planning and production. All details necessary for manufacturing and assembly are geometrically represented. Objects with MEP connections or structural mounting shall have exact representation of this in the geometry. Geometry, location and orientation is exact.</td>
</tr>
<tr>
<td>LoG Level 5</td>
<td>The object can be used for, but not limited to, e.g. FM, operation, maintenance and development. Geometry, location and orientation is exact, representing the physical construction.</td>
</tr>
</tbody>
</table>

B.3.2 Process status code and Level of Geometry (LoG)

Process status code and Level of Geometry (LoG) does not have similar purposes.

LoG describes requirements to geometry for the individual object classes (e.g. slabs, doors, valves, lighting fixtures). The process status code communicates how far the object is matured in the decision and quality control processes to satisfy the requirements (both geometry and information)
C. Appendix C – Details related to requirements for principal / core disciplines

C.1 Details requirements for fire safety in model

There will mainly be four groups of parameters (requirements / information) in a fire technical model. Below is a description of when the different groups are used and on what types of objects.

1. NOSSB_FireSeparator: The group specifies fire requirements for specific building components. The requirements mainly apply to horizontal and vertical dividing surfaces in and around a fire cell or fire section. There are typically requirements for walls, doors, windows, openings, slabs, roofs. The building element has either requirements for a given function as a fire barrier or requirements for a specific fire technical property. Objects on which these requirements are placed appear from the following list:
   a. Requirements for vertical surfaces, walls, and any horizontal doors / hatches / windows are placed on:
      i. IfcWall
   b. Requirements for doors / hatches are placed on:
      i. IfcDoor
   c. Requirements for windows are placed on:
      i. IfcWindow
   d. Requirements for horizontal surfaces, slabs / roofs / foundations, and any horizontal doors / hatches / windows are placed on:
      i. IfcSlab
   e. Requirements that include all objects in an entire volume. Typical surface, cladding and requirements for non-combustible insulation are placed on:
      i. IfcSpace
      ii. IfcZone
      iii. IfcBuildingElementProxy.USERDEFINED.FireSeparator

As an alternative to a. - e., all requirements for horizontal and vertical surfaces can be placed on på IfcBuildingElementProxy.USERDEFINED.FireSeparator.

2. NOSSB_FireSectionAndCompartment: The group specifies requirements that affect technical installations or building elements in an entire fire cell or an entire fire section. This can be, for example, requirements for sprinklers, guidance systems and fire alarm systems. Objects on which these requirements are placed appear from the following list:
   a. Requirements for fire cell / fire section are placed on
      i. IfcSpace
      ii. IfcZone
      iii. IfcSpatialZone

3. NOSSB_FireInformation: The group specifies conceptual requirements that apply to the entire structure as well as the fire concept, especially with regard to escape and rescue. Objects on which these requirements are placed appear from the following list:
   a. Requirements for rooms, floors and buildings are placed on
      i. IfcBuilding
ii. IfcBuildingStorey
iii. IfcSpace
iv. IfcZone
v. IfcSpatialZone

b. Objects used to provide information about fire concept without being directly related to a room, for example, the direction of escape or suggestions for the location of a fire alarm panel are placed on
i. IfcBuildingElementProxy.USERDEFINED.FireInformation

4. NOSSB_FireEscapeAndRescue: The group of requirements applies to whole individual components especially with regard to escape and rescue. Objects on which these requirements are placed appear from the following list:
   a. Requirements for doors in escape routes or points of attack for the Fire Department are placed on:
      i. IfcDoor
      ii. FireSeperator
      iii. FireEscapeAndRescue
   b. Requirements for components in a volume such as escape width etc. are placed on:
      i. IfcSpace
      ii. IfcZone
      iii. IfcSpatialZone

C.2 Details requirements for acoustics in model
There will mainly be four groups of parameters (requirements / information) in an acoustic model. Below is a description of when the different groups are used and on what types of objects.

1. NOSSB_AcousticSeparator: The group specifies sound technical requirements for specific building elements. Requirements mainly apply to horizontal and vertical dividing surfaces. There are typically requirements for walls, doors, windows, penetrations, slabs, roofs. The building element has either requirements for a given function as a sound insulation divider or requirements for a specific sound technical property. Objects on which these requirements can be placed appear from the following list
   a. Objects for requirements for walls, and any horizontal doors / hatches / windows are placed on:
      i. IfcWall
   b. Objects for requirements for doors / hatches are placed on:
      i. IfcDoor
   c. Objects for window requirements are placed on:
      i. IfcWindow
   d. Objects for requirements for decks / roofs / foundations, and any adjacent doors / hatches / windows are placed on:
      i. IfcSlab

   As an alternative to a. - e., all requirements for horizontal and vertical surfaces can be placed on på IfcBuildingElementProxy.USERDEFINED.AcousticSeparator.
2. **NOSSB_AcousticSectionAndCompartment**: The group specifies requirements that apply to rooms or building volumes. This can be, for example, requirements for reverberation time, maximum sound level, equivalent sound level, and more. Objects on which these requirements can be placed appear from the following list:
   a. Objects for reverberation time and sound level requirements are placed on:
      i. IfcSpace
      ii. IfcZone
      iii. IfcSpatialZone

3. **NOSSB_AcousticInformation**: The group specifies conceptual requirements that apply to the entire structure. Objects on which these requirements can be placed appear from the following list:
   a. Objects for requirements for rooms, floors and buildings are placed on:
      i. IfcBuilding
      ii. IfcBuildingStorey
      iii. IfcSpace, IfcZone
      iv. IfcSpatialZone
   b. Objects used to provide information about sound concept without being directly related to a room. Such as sound power level for equipment are placed on:
      i. IfcBuildingElementProxy.USERDEFINED.AcousticInformation

### C3 Details for requirements for process status codes in premise discipline models

As premise subjects do not own constructable objects in the model, a process status code for premise discipline objects will state the validity of the requirements that are set. Requirements from principal / core disciplines are in principle minimum solutions that must be satisfied for the structure to be in accordance with current regulations and standards.

In order for principal / core disciplines to work with the same methodology as other disciplines, it is recommended to use the Entrepreneurs’ Association Building and Construction (EBA) model maturity index (MMI) codes unless otherwise agreed. If the project uses a different coding system, the code must still be entered in the properties under NOSSB_Process.

Suggestions for division and detailing of EBA’s MMI guide for other subjects can be found in Appendix B. For premise disciplines, the layout is simplified. Use of existing / new codes is agreed in the project based on cost / benefit. The MMI code for principal / core disciplines will be ahead of the other disciplines as this is a design basis that is necessary for the disciplines to be able to increase the maturity of their objects.

<table>
<thead>
<tr>
<th>MMI code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMI200</td>
<td>Model by end of sketch project / Undetermined objects in preliminary project or detail project.</td>
</tr>
<tr>
<td>MMI350</td>
<td>Preliminary project</td>
</tr>
<tr>
<td>MMI400</td>
<td>Design basis detail project</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>APPROVAL DATE</th>
<th>28.06.2021</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPROVED BY</td>
<td>Anders Fylling</td>
</tr>
<tr>
<td></td>
<td>direktør faglig ressurssenter</td>
</tr>
<tr>
<td>REVISION NUMBER</td>
<td>1</td>
</tr>
<tr>
<td>PROFESSION AND METHOD RESPONSIBLE UNIT</td>
<td>FPROS</td>
</tr>
</tbody>
</table>