ETIM International

The international classification standard for technical products

ETIM Modelling Classification Guidelines

Version: 2.0

Status: Published

Author: Benjamin Sjölander

Date: 09/12/2025

Table of Contents

1. Introduction

In many industrial sectors, standardized product classification and information management are essential for seamless communication and data exchange among stakeholders. ETIM Modelling Class (ETIM MC) extends the ETIM classification standard, offering tools for parametric modelling that enhance efficiency, reduce errors, and support sustainability in design and construction. ETIM MC are standardized digital definitions, linked to ETIM product Classes, that describe product geometry using parametric features and 2D reference drawings. This foundational structure enables consistent, neutral digital representation of product geometry, supporting scalable and interoperable use in BIM and data management systems.

ETIM MC enables manufacturers, BIM users, wholesalers, and data distributors to manage and share standardized geometrical product data efficiently. It supports the creation of reusable, generic models that users can adapt to be manufacturer-specific through data exchange. This supports a wide range of use cases, from creating neutral 3D templates for BIM workflows to structuring product geometry in PIM, MDM and data exchange platforms. This promotes interoperability across systems, reduces complexity, and supports compliance with public procurement requirements.

ETIM MC offer fundamental advantages for digital construction and product data management, addressing key industry needs. Its generic, neutral design provides brand-independent shapes and dimensions. Parametric templates offer reusable object definitions for scalable BIM modelling. Furthermore, ETIM MC ensures seamless integration with ETIM technical product data, which can be utilized throughout the design and modelling process. This leads to uniformity in manufacturer-neutral geometry, promoting standardization across all markets and tools. Interoperability is boosted through seamless integration with BIM tools and open standards like IFC, enabling efficient cross-platform data exchange. The standard’s scalability supports a wide range of applications, from individual product components to complex systems, and its efficiency reduces the need for custom BIM content creation, streamlining workflows and accelerating digital project implementation. Ultimately, ETIM MCs streamline digital construction and reduce modelling complexity by connecting supply chain data with engineering data, futureproofing processes by aligning with evolving digitalization trends and increasing public sector requirements.

The ETIM MC Guidelines are designed for, but not limited to:

- **Manufacturers:** To standardize product data and enhance integration with digital platforms.
- **BIM Specialists:** To create accurate and scalable models for project planning and execution.
- **Software Developers:** To design tools that support ETIM MC integration.
- **Industry Organizations:** To promote standardization and collaboration across stakeholders.

- **ETIM National Chapters & Expert Groups:** To contribute to the development and maintenance of ETIM MCs, based on market needs and industry requirements.

It is maintained via the ETIM Modelling Class Management Tool (MMT) and is accessible through the ETIM API for seamless integration into software and workflows. ETIM International oversees the development, maintenance, publication and promotion of the ETIM standards.

1.1. ETIM Classification license Info

The ETIM classification standard (product classification) and ETIM MC standard are made available under the Open Data Commons Attribution License:
http://opendatacommons.org/licenses/by/1.0/

Example attribution:
"This data is based on the ETIM classification model, provided under the Open Data Commons Attribution License by ETIM International."

This license allows users to freely copy, distribute, and use the data, provided proper attribution is given to ETIM International. This ensures that the standard remains openly accessible while maintaining recognition of its source and management.

1.2. Purpose of the document

These guidelines describe a set of principles, rules and best practices for creating, managing, and integrating standardized ETIM Modelling Classes to ensure consistency, quality and interoperability across MDM/PIM, BIM software and other digital workflows. The objective is to support interoperability, accuracy, and scalability in product modelling, enabling seamless collaboration and data exchange across sectors and applications. By following these guidelines, stakeholders can contribute to a more cohesive, efficient, and innovative approach to product modelling and classification.

1.3. Key differences between ETIM Product Classes and Modelling Classes

ETIM includes two class types with distinct roles:

- **Product Classes (EC):** Defines the technical and functional characteristics of products for identification and comparison. EC is an abbreviation for **E**TIM Product **C**lass.
- **Modelling Classes (MC):** Provides a standardized geometric representation for BIM and configuration tools. MC is an abbreviation for ETIM **M**odelling **C**lass.

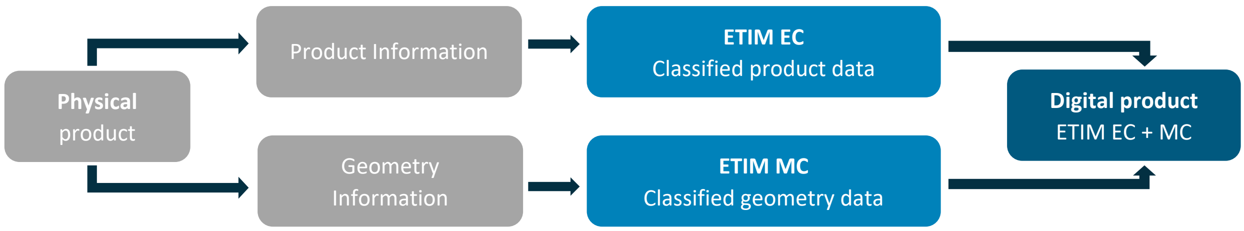


Figure 1 From physical product to digital product.

Comparison summary:

Table 1 Key differences between ETIM Product Classes and Modelling Classes.

Aspect	Product Classes (EC)	Modelling Classes (MC)
Purpose	Technical classification	Standardized geometric definition
Features	Functional and technical	Dimensional and connection-based
Drawing	No	Yes
Linkage	Standalone	Linked to at least one EC class
Use focus	Commerce, data exchange, data management	Geometry modelling, data management, BIM integration

Modelling Classes complement Product Classes by adding geometry to support digital workflows, without replacing or modifying the Product Classification logic.

Key characteristics introduced by ETIM MC:

- **Reference drawings:** Each MC includes a standardized 2D drawing linking dimensions directly to features for clear and consistent interpretation.
- **Connection Types and PortCodes:** Modular integration of Connection Type Classes and PortCodes defines how products connect, supporting consistent behaviour across platforms.
- **Dedicated Modelling Groups:** MCs are organized into Modelling Groups (MGs) by product type and sector, maintaining separation from EC hierarchies and avoiding classification conflicts.
- **Enhanced parametric focus:** MCs support scalable, reusable geometry compatible with various design and construction tools.

1.4. Standards alignment overview

The ETIM Modelling Class Guidelines are developed with references to internationally recognized standards in related areas such as classification, product data modelling, and BIM workflows. A detailed compliance overview is included in chapter *8. References and standard alignments*.

ETIMs approach is to align with principles from established external standards where they enhance clarity and consistency within the scope of ETIM MC’s modelling purpose. The pragmatic alignments ensures that the guidelines are both practical and grounded in already established conventions.

Key areas of alignment:

- **Technical documentation**
  Principles for reference drawings, dimensioning and presentation.

- **Data structure and BIM**
  The object-oriented structure and data management principles are consistent with established standard conventions.
- **Semantic interoperability**
  The guidelines are partially aligned with standards for digital product data and terminology.

Interoperability with the IFC schema (ISO 16739) is supported through ETIM's integration with the buildingSMART Data Dictionary (bSDD). Future updates to these guidelines may adopt additional ISO practices to further enhance interoperability and adoption.

A vocabulary appendix is not included in the guidelines, as all terminology is governed through the Classification Management Tool (CMT) and Modelling Management Tool (MMT), which implement semantic versioning, role-based workflows, and change control aligned with terminology management principles.

1.5. Modelling focus and boundaries

While ETIM MC provides a robust, scalable framework for standardized modelling, it intentionally imposes certain limitations to support interoperability, neutrality, and low-detail modelling across diverse platforms and industries.

Table 2 Known limitations of the ETIM Modelling Classification standard.

Limitation	Description
Low Level of Detail (LOD)	MCs represent schematic, not manufacturing-level, geometry.
No brand-specific geometry	Exclude logos, shapes, or other manufacturer-specific features from the modelling class.
Simplified flow & behaviour representation	Flow, electrical, or thermal behaviour is represented symbolically only.
Rigid modelling of flexible components	Cables, hoses, etc., are shown as fixed forms with no dynamic bending.
No modular assembly logic	MCs define single objects with limited modularity. MCs can have a placeholder for different types of connections, but cannot have references to components that have a specific MC.
Limited support for unique custom objects	Highly specific or one-off designs are not covered unless generically abstracted.
Static coordinate system	All models use a fixed global XYZ system (right-handed, Z up); no transformations are applied.
Dependency on linked class versions	MCs rely on versioned EC and CT links; changes to those require new MC versions.

Redundant dimensional features between EC and MC	Overlap in dimensions between ECs and MCs (e.g., width, height) still exists due to system dependencies.

Note: Any limitations are by design and support broad compatibility, efficiency, and neutrality. Future updates may address some of these constraints based on evolving needs. Future versions of ETIM MC may address these limitations or introduce expanded functionality based on evolving market requirements and technological developments.

2. ETIM Modelling Classification data model

The ETIM Modelling Class (MC) data model is based on the core ETIM classification structure, with dedicated extensions to support geometric representation and engineering applications. This section outlines its general structure, highlights distinctions from Product Classes (ECs), and references key resources for data exchange and API integration.

2.1. General structure

The ETIM Modelling Class (MC) framework extends the core ETIM classification model by incorporating structured support for geometric representation and engineering semantics. It builds upon the existing entity model of ETIM and introduces formal linkage mechanisms to enable low-detail parametric modelling suitable for BIM and digital workflows.
This structure forms the semantic and geometric foundation needed to enable scalable data exchange and consistent behaviour across BIM tools, PIM platforms, and CAD content libraries.

Core entities
All entities in the ETIM MC framework are uniquely identified and managed through two platforms:

- The Classification Management Tool (CMT): for features, values, units and product classes.
- The Modelling Management Tool (MMT): for Modelling Classes, Modelling Groups, Connection Types, and associated drawings.

Table 3 Overview of core ETIM MC data components and their roles in the classification structure.

Type	Description
Product Classes	Define product typologies based on function. **Example:** EC010958 – Membrane pressure expansion vessel.
Modelling Groups	Group related Modelling Classes that share common characteristics or functionalities within a product category. **Example:** MG000033 – Pressurisation
Modelling Classes	Define parametric geometric templates, including a unique ID, 2D reference drawing, and assigned features. **Example:** MC000105 – Membrane pressure expansion vessel, vertically suspended, top connection

Features	Structured property definitions, including dimensional, options, or technical attributes. **Example:** EF000065 – Inner diameter.
Values	Type of allowable inputs in a feature with either: logical, alphanumerical, numeric or range-based data. **Example:** EV0020109 – DN 6 (Type: Alphanumeric).
Units	Measurement systems, metric and imperial, applied to features using numerical inputs.
Connection Type Classes	Define modular connection parameters (e.g., flanges, threads) to standardize port handling across objects.

The interaction between these entities, such as the linkage between Modelling Classes and Product Classes, or the use of shared features, is detailed in Section 2.2.3.

2.2. ETIM Modelling Class framework overview

2.2.1. Modelling Groups

Modelling Groups (MGs) organize MCs into logical sets based on product type or application area. Each group has a unique identifier and code that follows a fixed format.

- **Code Format:** Each group is assigned an "MG" prefix followed by a six-digit number (e.g., MG000053).

MGs provide administrative structure for managing Modelling Classes independently from ETIM's EC group structure. They help maintain classification clarity and avoid structural conflicts between ECs and MCs. Modelling Groups are centrally maintained in the MMT and are not tied to product hierarchy or market-specific categorization.

2.2.2. Modelling Class

ETIM MCs define standardized, parametric representations of products within the ETIM framework. Each MC captures the essential geometric and technical attributes of a product type, based on its associated EC.

Key elements of each MC include:

Table 4 Core elements defining an ETIM Modelling Class and their role in standardized geometric representation.

Type	Description
Unique identification	Every MC is assigned a unique class code (e.g. MC000038) for version control and traceability.
Parametric dimensions	Physical characteristics such as length, width, height, and diameter are defined as features for accurate, scalable modelling.
Reference drawing	A standardized 2D drawing illustrates how dimensions relate to geometry, ensuring consistency in model interpretation.
Connection features	Interfaces such as threads or flanges are described using linked Connection Type Classes (CTs) and PortCodes.

Engineering features	Additional attributes (e.g., flow rate, load capacity) provide context for modelling and specification.
Distinct object types per product class	A single Product Class may be represented by multiple MCs to account for geometric variants (e.g., rectangular vs. circular).

2.2.3. Connection Type Classes

Connection Type Classes (CTs) define standardized interfaces for how objects connect within a system. These may include common connection forms such as threads, flanges, sockets, or plain ends.

CTs allow connection geometry and semantics to be defined independently of the object body, enabling modular and consistent modelling across classes.

Key principles:

- **Centralization:** CTs capture all relevant data for a connection type, such as dimensions, port shapes, and types, without duplicating content in the MC.
- **Reusability:** A single CT can be linked to multiple MCs, reducing redundancy and simplifying updates.
- **Backward compatibility:** Existing MCs without CT references remain supported, but new MCs are expected to use CTs where applicable.

CTs are governed through the same RFC process as MCs and are version-controlled in the MMT.

Linking Connection Types to Modelling Classes

By linking CTs to MCs with unique PortCodes for each connection point within the model, ensures accurate, scalable handling of complex configurations while supporting a flexible efficient product representation and maintaining data integrity.

Core concepts:

- **PortCodes:** Numeric identifiers (e.g., 1, 2, 3) that define individual connection points. Each PortCode is associated with at least one CT class.
- **Multiple CTs per MC:** A single Modelling Class may include several CTs, each assigned to a PortCode to represent multiple connection interfaces.
- **Automated updates:** When a CT is revised, changes automatically update all linked MCs through the MMT, ensuring consistency and minimizing manual edits.

2.2.4. Features and values overview

Features and values used in ETIM Modelling Classes and Connection Type Classes are part of the central ETIM classification system and are also shared with ETIM Product Classes.

Semantic reuse is encouraged, where appropriate, the same feature and value codes are applied across classes to ensure consistency and reduce redundancy.

All features and values are managed within the CMT and must receive administrative approval before becoming available in the MMT.

Geometry handling:
If dimensional features (e.g., width, height) are already defined in the EC, the MC must still define geometry independently according to MC modelling rules. MCs do not inherit or rely on EC geometry, ensuring neutral, parametric modelling tailored for BIM and CAD environments.

Additional Feature types:
ETIM MC also supports two advanced feature types for spatial and engineering use:

- **Coordinate (C):**
  Represents spatial points in X, Y, and Z coordinates using signed decimal values. Supports accurate positioning of geometric elements.
- **Matrix (M):**
  Defines structured data as key-value sets enabling representation of complex logic or lookup tables. This feature type has specific use for engineering purposes by utilizing a “IF x THEN y” approach, enabling possibilities such as graph-creations and analysis.

These feature types are supported in the ETIM API for implementation and data exchange.

2.2.5. Unit policy

ETIM MCs require both metric and imperial units. All dimensional and geometric features and Coordinates (CA), must define values in millimetres (mm) and inches (in). Angular features must use degrees (°).

Each feature in the MC feature table must explicitly specify both units. Units apply only at the data level and must be consistent across the class to ensure clarity and interoperability.

2.3. Governance and organizational structure

The ETIM Modelling Classification standard is governed through structured processes that ensure consistency, scalability, and transparency across all entities. Governance mechanisms differ based on entity type and usage context but follow shared principles of traceability and controlled versioning.

Class entities

- MCs and CTs are governed through the formal Request for Change (RFC) process.
- Structural changes (e.g., geometry, features, PortCodes) require an RFC and are version-controlled.

- Each approved change results in a new version, managed in the MMT and made available via the ETIM API.
- Translations of MCs and CTs also follow the RFC process and are version-bound.

Supporting entities

Entities such as Features, Values, Units, and Modelling Groups are centrally maintained but not versioned independently.

- Features and values are managed in the CMT and reviewed by ETIM International.
- Modelling Groups are maintained in the MMT.

Table 5 Governance responsibilities

Entity	Managed in	Governed by	RFC required	Versioned	Notes
Modelling Classes	MMT	ETIM International	Yes	Yes	Full RFC process and controlled release
Connection Type Classes	MMT	ETIM International	Yes	Yes	Governed like MCs
Features / Values / Units	CMT	ETIM International	No	No	Internal review, not versioned
Modelling Groups	MMT	ETIM International	No	No	Translations by national chapters

Language and translation governance

- The system language, ETIM English, is maintained by ETIM International for all entities.
- Translations are managed by national ETIM chapters and are integrated at different levels depending on entity type.

2.3.1. Governance bodies

The ETIM Modelling Classification standards governance, development and maintenance are overseen by a governance structure that operates within the broader ETIM organization. This document covers the two unique instances specifically formed for the ETIM MC standard. Read more about the ETIM organization structure in the main ETIM Guidelines: ETIM Classification and format Documents.

- **ETIM MC Steering Committee**
  This committee, consisting of representatives of ETIM MC Full members, oversees the strategic direction of the development and promotion of the ETIM MC standard. It guides the long-term vision and ensures alignment with market needs. The Steering Committee reports with the ETIM Staff Office to the ETIM International Board. The Steering Committee has at least two meetings per year.

- **ETIM MC Technical Committee**
  The Technical Committee of ETIM MC is responsible for the technical content and the standardization process of ETIM MC. It reviews and decides on complex proposed Request for Change (RFCs) or when the issues implications have a larger impact beyond the specific RFC. The Technical Committee consists of ETIM MC Full members. A representative from the ETIM International Staff Office is the chair of the Technical Committee and reports to the Steering Committee. The Technical Committee typically has four meetings per year, with additional meetings scheduled as needed.

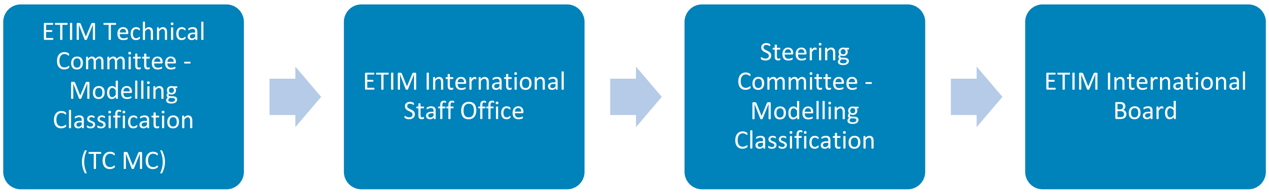


Figure 2 Governance structure of the ETIM Modelling Classification standard

2.4. Relationships

The ETIM Modelling Class system maintains structured relationships between core entities to ensure semantic consistency, modularity, and reliable versioning across platforms.

Functional linking

- Every MC is linked to at least one EC to reflect the product's intended function.
- A single EC may reference multiple MCs to support different geometric forms for product variants.
- These links are version-controlled and stored as part of the MC’s metadata.

Geometric and interface modularity

- MCs define parametric geometry with 2D reference drawings that visually indicate dimensional features.
- CTs group reusable geometry parameters for connection interfaces (e.g., threads, flanges) and are linked to MCs using PortCodes.
- This separation allows consistent port modelling while minimizing redundancy.

Data consistency and integration

- Features, Values, and Units are centrally maintained in the CMT and reused across ECs and MCs.
- Semantic consistency is enforced across classes and systems.
- Any structural change to an EC or CT triggers the creation of a new MC version to maintain data integrity.

Figure 3 illustrates how core entities within the ETIM Modelling Classification relate to each other and to the existing ETIM Product Classification.

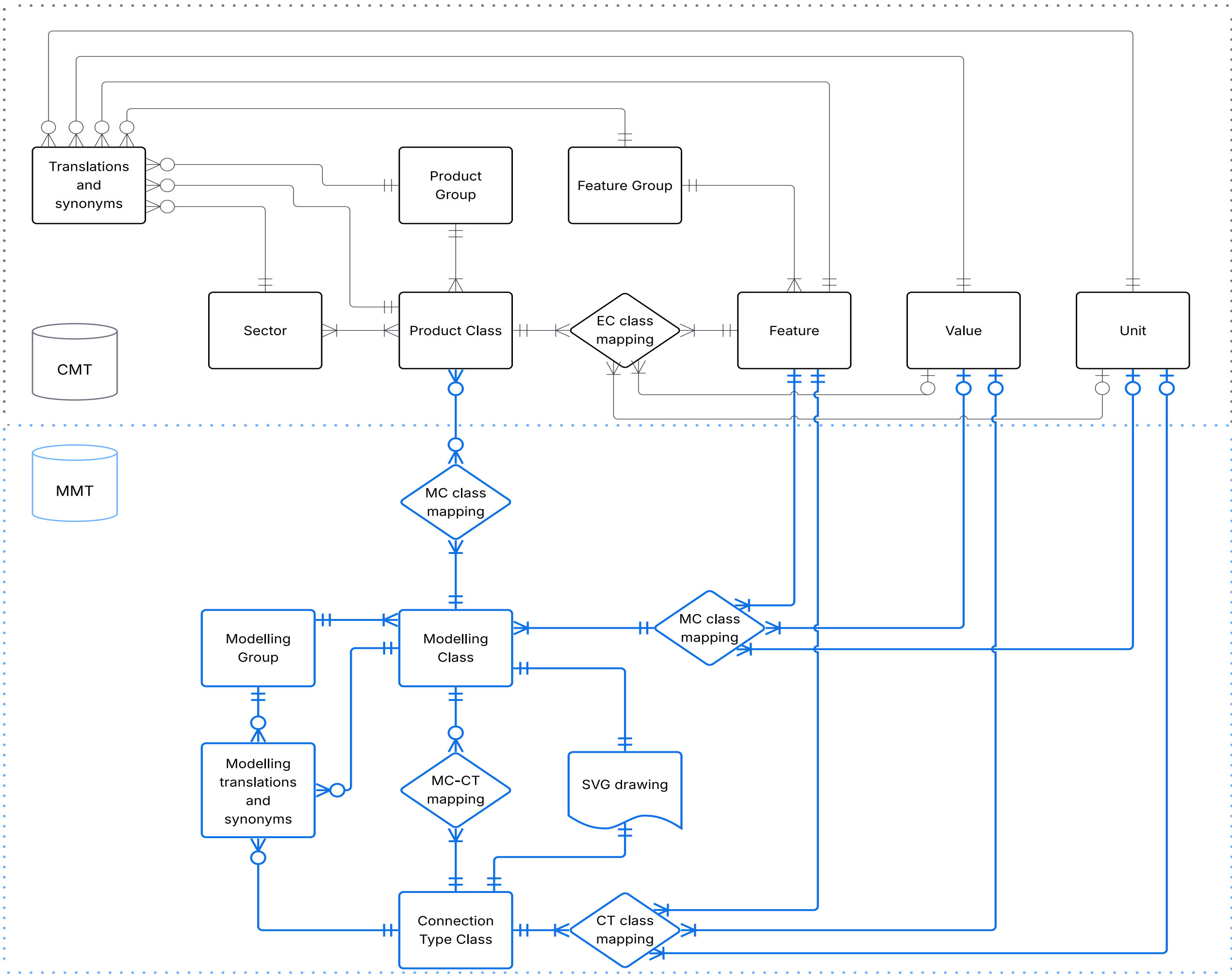


Figure 3 Overview of the relationship between ETIM Modelling Classification and ETIM Product Classification.

2.5. Reference to documentation on exchange formats and ETIM API

These resources are essential for standards-compliant implementation of ETIM Modelling Classes in digital workflows. Access to these documents is available via the:

ETIM xChange format documentation (link)

- Defines the structure and content of exchange files for ETIM data, including Product and Modelling Classes.
- Includes JSON schema definitions, field mappings, and example files.
- Supports both batch and incremental data synchronization between systems.

ETIM API Portal (link)

- Offers programmatic access to the ETIM classification data.
- Provides endpoints for:
    - Retrieving Modelling Class metadata

- Accessing features, values, and drawings
- Navigating version histories and linked entities

- Includes authentication instructions, usage examples, and integration guides.

Documentation for both is available via the ETIM International downloads page.

Note: Conformance to the xChange format and API specifications is essential for successfully implementing the ETIM Modelling Classification standard across various platforms.

2.6. IFC and bSDD interoperability

ETIM Modelling Classes are available in the buildingSMART Data Dictionary (bSDD), enabling semantic interoperability with IFC (ISO 16739) through linked identifiers. Although defining native IFC object schemas is out of ETIM MCs scope, its presence in bSDD supports semantic mapping to IFC concepts. This allows BIM tools that use IFC-based object representations to automate object mapping, enrich data semantically, and validate classifications during model authoring.

3. Versioning and change management

This chapter outlines how versioning is managed within the ETIM Modelling Classification, including structured release phases, the formal Request for Change (RFC) workflow, and the rules governing version relationships between Modelling Classes and linked classification entities.

3.1. Versioning process

The ETIM Modelling Classification follows a structured release cycle aligned with the ETIM Product Classification. This ensures data consistency, predictable updates, and coordinated integration across systems.

Triggering event

- The official release of the ETIM Product Classification triggers the start of the Modelling Classification update cycle.
- This release typically occurs every 24 months and includes finalized changes to Product Classes.

Release phases
The MC versioning process includes three formal stages and one ongoing phase:

1. **MC RFC Lockdown**
   Once the EC release is finalized:
   - No new RFCs are accepted.
   - Open RFCs for MCs and CTs are reviewed.

- RFCs are either approved, returned for revision, rejected or withdrawn.

2. **MC Release**
   - Approved changes are officially published.
   - Updates may include new MCs, revised features, and class version updates.
   - Released versions are made available through the MMT and ETIM API.

3. **Dynamic Release**
   - After the main release, newly approved RFCs may be published provisionally.
   - These updates are accessible via the MMT and API for early use as a pre-release.

This release model balances controlled versioning with flexibility for early adoption and iterative improvement.

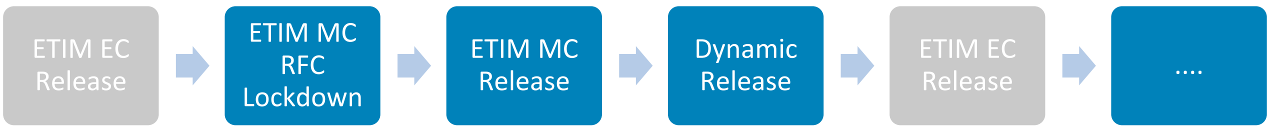


Figure 4 ETIM MC Release process

3.2. Change management

Modelling Classes and Connection Type Classes are updated through a formal Request for Change process. This process ensures controlled evolution of the classification model through collaboration, transparency, and version control.

3.2.1. Request for Change workflow

The Request for Change (RFC) process allows ETIM members to propose new Modelling or Connection Type Classes or to update existing ones. It ensures traceability, collaboration, and alignment with the ETIM MC data model.

The workflow consists of three main phases:

1. **Initial request**
   Initiated by a stakeholder (e.g., manufacturer, engineer, data distributor, or other ETIM member).
2. **Preparation & submission**
   The responsible national ETIM chapter prepares the RFC, including required dimensional drawings and supporting documentation.
3. **Evaluation & decision**
   The ETIM International Staff Office reviews the RFC for consistency with modelling guidelines and versioning policies. The Staff Office has the mandate to document and make necessary changes in an RFC if needed.

All RFCs must include a clear justification. Drawings and reference products are mandatory where applicable. ETIM Members may ask the Staff Office to act as RFC owner and directly draft a new Modelling Class request in the MMT's Proposal stage before Initial Control.

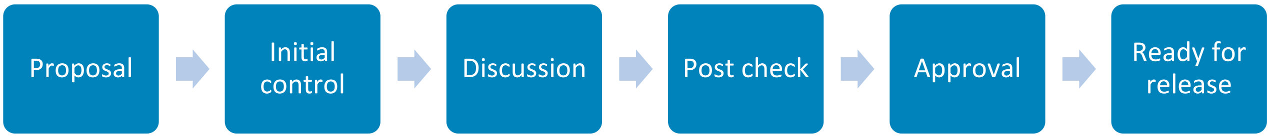


Figure 5 RFC process overview diagram

RFC workflow stages

Table 6 Stages in the RFC workflow and responsible parties

Stage	Description	Responsible party
Proposal	RFC is prepared in the MMT. The RFC owner ensures all submission requirements are met.	ETIM MC Chapter, ETIM International Staff Office
Initial Control	Staff Office checks completeness and compliance. Incomplete RFCs are returned with feedback.	ETIM International Staff Office
Discussion	60-day open comment period for ETIM members. Contested changes may trigger a new review.	ETIM MC Chapter, ETIM International Staff Office
Post Check	Final review and feedback review. Minor fixes are applied; major issues may return RFC to earlier stages.	ETIM International Staff Office
Approval	RFC is accepted and scheduled for release.	ETIM International Staff Office
Release	Changes are made available via the Dynamic Release and versioned in the next scheduled ETIM MC release.	ETIM International Staff Office

Multiple RFCs and conflict resolution

Multiple RFCs may be submitted for the same class if they address separate aspects. Overlapping proposals are reviewed for consistency and may be consolidated or escalated to the Technical Committee if conflicts arise.

Multiple RFC resolution rules:

Table 7 Conflict resolution strategies for overlapping RFCs

Stage	Trigger/Criteria	Key actions & outcomes
Initial screening	Overlapping or conflicting proposals	Staff Office flags issues and initiates deeper review
Parallel review	RFCs affect different aspects but only one affect the drawing	RFCs proceed independently through review

Merging conflicts	RFCs overlap but are not mutually exclusive	RFCs may be merged; submitters are notified and encouraged to collaborate
Escalation	Conflicting RFCs (e.g., incompatible geometry)	Escalated to Technical Committee or expert mediator; submitters provide justification
Decision & communication	Final resolution achieved	Staff Office publishes rationale, applies accepted changes, and returns conflicting RFCs as needed

3.2.2. Request for Change requirements

To ensure consistency, accuracy, and usability, all RFCs must meet specific input requirements before being accepted into the review process. These criteria standardize how Modelling Classes and Connection Type Classes are proposed or modified.

> **Note:** If a drawing is to be created by the Staff Office, a technical drawing and at least two reference products must still be provided at the time of submission.

RFCs are submitted via the MMT:

- For new Modelling Classes: use the "*New class…*" option found in the "*Actions*" button under the "*Classes*" tab.
- For existing classes: select the class and use the green plus icon in the *RFC panel* under the version list.

Table 8 Required fields for submitting an RFC in MMT

Type	**Description**
Class type *(for new classes)*	Choose if the requested new class is a Modelling Class or a Connection Type Class.
Description	A clear, neutral title describing the object's basic characteristics.
Group	The Modelling Group most appropriate for the requested class.
Requester company	Organization or company submitting the RFC.
Requester name	Individual responsible for the RFC submission.
Requester country	Country of origin for the request.
RFC title	Short, descriptive title summarizing the change.
RFC type *(for existing classes)*	Nature of the change (e.g., "Rework class", "Add values").
RFC Description	Detailed explanation and justification in English.
Product classes	Relevant ETIM Product Classes (latest version only), linked under Linked Classes.
Connection Types	Linked Connection Type Classes with assigned PortCodes (if applicable), managed under Linked Classes.

Drawing	Drawing that meets ETIM format requirements.
Features	List of parametric features with drawing codes, metric and imperial units. At least two values must be added for alphanumeric option lists.
RFC Reference product	Reference images are uploaded in the RFC version view to illustrate the scope of the class and prove added features relevance. They should be as brand-agnostic as possible. At least two reference products are mandatory and recommended to be from different manufacturers. Reference products need to have the source link attached.
Reference product	Additional supporting images, uploaded in the Class version view.

3.3. Versioning relationships

To ensure data stability and backward compatibility, every Modelling Class explicitly references the version of each linked Product Class and Connection Type Class at the time of its creation.

These version links are fixed and form part of the MC's version identity.

Key Principles:

- Updates to an EC or CT do not retroactively alter existing MC versions.
- Any change in a linked class that impacts semantics, or structure will be integrated in a new MC version, created through the RFC process.
- Version dependencies are recorded and managed in the MMT by the ETIM International Staff Office before each official release.

Example:

Table 9 Versioning relationship example; New versions of EC and/or CTs triggers a new MC version.

Modelling Class	Linked Product Class	Linked Connection Type Class
MC000010 – V2	EC003034 – V9	CT000001 – V1
MC000010 – V3	EC003034 – V10	CT000001 – V2

Each MC version is self-contained and stable. Changes to linked classes always result in a new MC version to preserve historical integrity and traceability.

Created 3D template objects based on and ETIM Modelling Class should refer to the specific Modelling Class it was based on, including MC ID, version number and release number.

4. Quality assurance and validation practices for implementers

To maintain consistency, usability, and integrity across ETIM Modelling Classes, a set of quality assurance and validation practices is recommended throughout the modelling and governance lifecycle.

Table 10 Recommended quality assurance and validation measures

Type	Description
Internal reviews	Each organization (e.g., ETIM chapter, data distributor or modeller) should routinely check its own Modelling Class data, drawings and 3D template objects to ensure alignment with ETIM MC guidelines.
Peer reviews	Independent verification by domain experts or ETIM MC users to confirm that semantics and drawing conventions are correctly applied.
Automated validation	Use built-in checks, software tools or the ETIM xChange validation tool for validation of data and compliance.
User feedback	Establish feedback loops with end-users to capture real-world issues and identify improvements.
Audit trails	Detailed tracking of RFC submissions, decisions, and implementations within the MMT for transparency and accountability.

5. Drafting and drawing practices

Parameter reference drawings are a core component of the ETIM MC standard. These standardized 2D illustrations define the spatial location of parametric dimensions. Their purpose is to ensure consistent, clear representation of product geometry across the classification.

Purpose and principles
A parameter reference drawing serves as a visual blueprint, indicating the position and function of each dimensional parameter in relation to the modelled object. It is not intended to prescribe the Level of Detail (LOD) for any specific software or use case, including BIM environments. Instead, it provides a consistent dimensional reference that remains compatible across various applications, platforms, and LOD requirements. In BIM workflows, this supports integration around LOD 200 and ensures cross-platform usability without requiring manufacturer-specific geometry.

Key principles for reference drawings:

- **Recognizability:** Drawings must clearly resemble the generic form of the product to support easy identification.
- **Neutrality:** Geometry must avoid references to specific brands or proprietary shapes.
- **Simplicity over completeness:** Dimensional complexity should be minimized. Where needed, split products into multiple MCs to maintain clarity and usability.

5.1. Drawing structure and format

The drawing structure in ETIM MC ensures that all visual documentation is both technically accurate and consistently presented. The following subsections describe the expected level of detail, formatting standards, and template usage that support cross-platform interoperability and maintain clarity across applications.

5.1.1. Level of detail (LOD)

ETIM Modelling Classes are typically developed around LOD 200, representing products at a low to medium Level of Detail in accordance with international BIM standards.
This supports:

- Generic spatial representation for coordination and planning
- Symbolic, parametric geometry without fabrication-level detail
- Efficient data exchange across BIM, PIM, and coordination platforms

ETIM MC objects intentionally exclude high-resolution geometry, manufacturer-specific features, or detailed installation data. Instead, they prioritize compactness, interoperability, and clarity for scalable digital workflows.

Minimum dimensional requirements
Each Modelling Class must define the minimum set of dimensions required to describe its geometry. If covered by other features, such as centre-to-centre distances or symmetrical spans, redundant dimensions must be avoided.

The goal is to ensure unambiguous, compact geometry and reduce complexity.

5.1.2. Drawing format

Parameter reference drawings are stored as SVG (Scalable Vector Graphics) files in the MMT and are version-controlled per MC class. Only vector elements are allowed. Raster or pixel-based images must be excluded to preserve maintainability, editability, cross-platform clarity and scalability.
SVG is selected for its resolution independence, scalability and vendor-neutral format. It is based on XML and CSS, making it compatible with modern web standards.

Drawing naming convention: *[ModellingClasscodeID]_V[VersionNumber].svg*
Example: MC000123_V2.svg

5.1.3. Drawing template

ETIM MC reference drawings must use the latest version of the drawing template when creating new modelling classes. For existing classes, reference drawings must be updated to the latest template as part of any RFC.

Table 11 Drawing template legend

Type	Description
Modelling class code	To which modelling class the reference drawing belongs.
Drawing title	The drawing title shall always be the name of the modelling class.
Class version	The corresponding modelling class version.
Revision date	The date when the drawing was updated.
Projection	The standard projection format of an ETIM MC model.
ETIM logo	ETIM International's latest logo shall always be used in the drawing.
Centre extension	Centred 10 mm extension lines on all four sides.

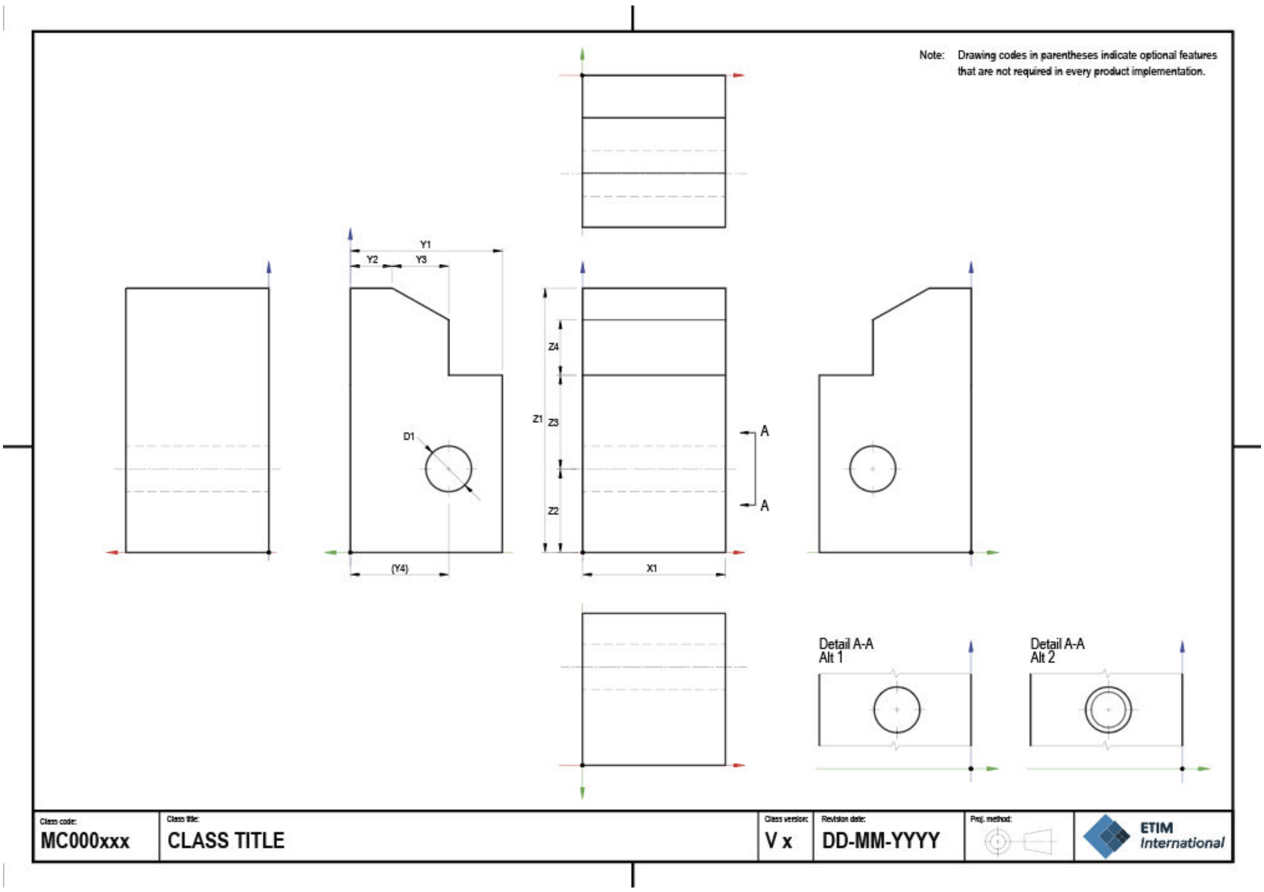


Figure 6 Drawing template with view arrangement, dimensions and object details.

5.1.4. Modelling Class name and drawing title

Modelling Class names in ETIM must follow a structured, rule-based naming convention that enables domain-relevant, semantically organized naming. This ensures clarity, consistency and interoperability for classification, data exchange and seamless integration into BIM, PIM, and digital workflows.

Names must be:

- Written in English
- Represent singular, manufacturer-neutral object types
- Matched exactly by the drawing title of the associated reference drawing
- Class names have a technical maximum length of 80 characters but must be kept within a reasonable limit to ensure readability and usability.

Naming structure
Each class name should be constructed using the following sequence:
[Object type], [Functional role/attribute], [System context], [Installation/mounting], [Form or variant]

The class name should follow a consistent order and fixed sequence without deviation. It starts with the base object type, which is the only mandatory component. All other components are optional and should be included to increase clarity and avoid ambiguities. The sequence continues with qualifiers that describe shape, function, application or variants. When multiple qualifiers are used, separate them with commas in descending order of specificity.

Table 12 Components used in Modelling Class names and their intended function.

Component	**Purpose**	**Examples**
Object type	Defines what the object is	Tee, cabinet, bend, sensor
Functional role	Describes behavior, function or connection style	Flanged, Threaded, indirect, intelligent
System context	Indicates technical application or domain	Pipe system, energy distribution, air duct
Installation	Describes mounting or placement	Wall-mounted, vertical, recessed
Form/variant	Subtype, distinguishing detail or physical shape.	With safety cage, bottom connection, rectangular, round

Table 13 Examples of valid Modelling Class names with corresponding structural components.

Valid class name	**Explanation**
Bend, air duct, rectangular	Object type – **Bend** System context – **air duct** Form – **rectangular**
Cap, threaded, pipe system	Object type – **Cap** Functional role – **threaded**

	System context – **pipe**
Sensor, movement, ceiling-mounted, alarm system	Object type – **Sensor** Functional role – **movement** Installation – **ceiling-mounted** System context – **alarm system**
Control element, intelligent, domestic switching, rectangular	Object type - **Control element** Functional role - **intelligent** System context - **domestic switching** Form – **rectangular**
Elbow unit, double vertical, busbar trunking system, Z-shape	Object type – **Elbow unit** Functional role – **double vertical** System context – **busbar trunking system** Form – **Z-shape**

Naming requirements:

- Names must be singular, neutral, and not brand- or product-specific.
- Use only standard ASCII characters. Avoid diacritics (e.g., é, ä, ø).
- Do not use slashes (/), ampersands (&), plus signs (+), or other special characters.
  - Do not include parentheses.
- Separate components with commas.
- Use hyphens only for compound adjectives (e.g., wall-mounted).
- Use sentence case: capitalize only the first word and abbreviations (e.g., IP).
  - Avoid abbreviations unless internationally recognized and unambiguous.
- The class name must not include a material (e.g., “Copper”)
- It should provide a shape if multiple variants exist (e.g., “Round”)
  - Use rectangular when the form can be either square or rectangular.
- It cannot only consist of a system context (e.g., “Fire alarm”)
- Avoid terms that are vague or that may apply to multiple object types.

5.2. Geometric object types

Most of the products relevant to ETIM Modelling Classes can be categorized into at least one of the following types of modelling shape:

- Symmetrical shapes
- Asymmetrical shapes
- Linear and path-based shapes
- Assemblies

A Modelling Class should aim to use the minimum number of parameters necessary to describe the standardized geometry, ensuring both efficiency and design flexibility.

However, when shapes or forms are not fully defined through drawing codes and features, they may be interpreted and added as external refinements to the Modelling Class.

Repetitive patterns

Repetitive patterns in MCs need to be handled with caution to ensure simplicity, neutrality, and usability of the ETIM MC standard. Repetitive patterns are allowed when the repeated shape has no variance in form. If the repetitions introduce new parameters, the number of repetitions need to be restricted. When multiple variations exist that can't be defined in a simple and generic way, multiple Modelling Classes are needed.

5.2.1. Symmetrical shapes

Many products conform to simple, standardized shapes that are ideal for parametric modelling:

- **Cylindrical**: Pipes, ducts, conduits, expansion tanks and cables.
- **Rectangular/Box-shaped**: Distribution boards, enclosures, radiators and building components like wall-mounted components (e.g., sockets and switches).
- **Flat/Panel-like**: Panels, floor plates, tiles and mounting plates.

Key recommendations:

- Use parametric dimensions for length, diameter, height, width and depth.
- Focus on minimal detail unless greater precision is necessary for functionality or compliance.
- Align shape orientation with industry conventions to support universal application.

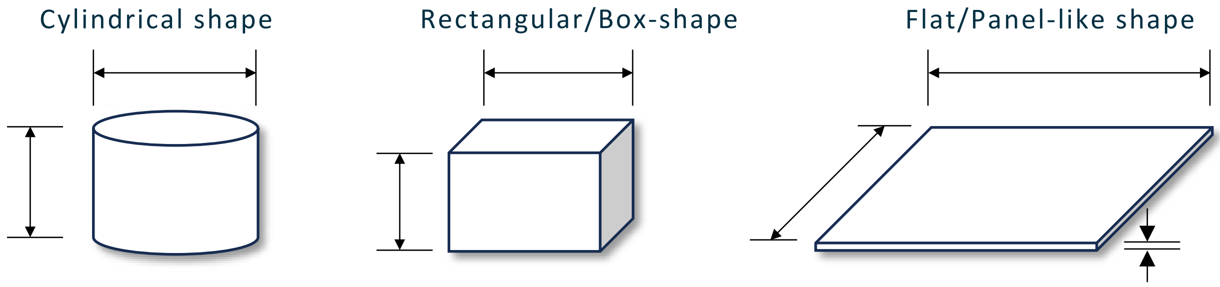


Figure 7 Example of symmetrical shapes

5.2.2. Asymmetrical shapes

Some simple shapes have more complex or custom forms that can't be defined as symmetrical geometry:

- Curved and angled shapes: Pipe elbows, HVAC duct bends and junction boxes.
- Specialized forms: Custom connectors, decorative building materials, or integrated assemblies with unique profiles.

Key recommendations:

- Include key dimensions like curvature radius, angle, and connection types.
- Simplify representations where detailed modelling adds low value.

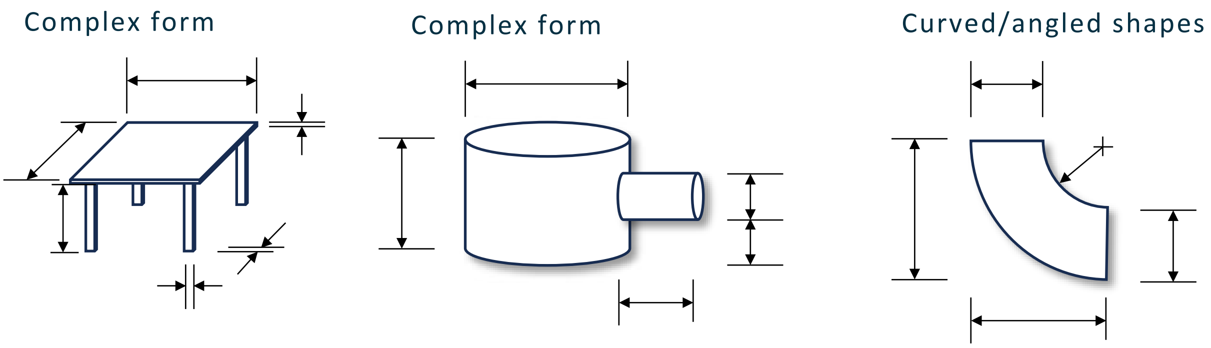


Figure 8 Asymmetrical shape examples

5.2.3. Linear and path-based shapes

These products are shaped along a path or direction, but their handling in MCs depends on how their segments are defined:

- **Ducts, pipes and tracks:** Bends and angled sections are typically separate products and should have their own distinct Modelling Classes (e.g., pipe elbows, duct bends).
- **Cables:** A cable remains the same product whether straight or bent. However, in BIM software, bends could be treated as components within the system but do not represent a unique product.

Key recommendations:

- **Cables:** Use standard attributes like length, diameter and optional radius for bending.
- **Ducts, pipes and tracks:** Create distinct Modelling Classes for straight and angled sections with attributes like angle, radius and connection types for bends.

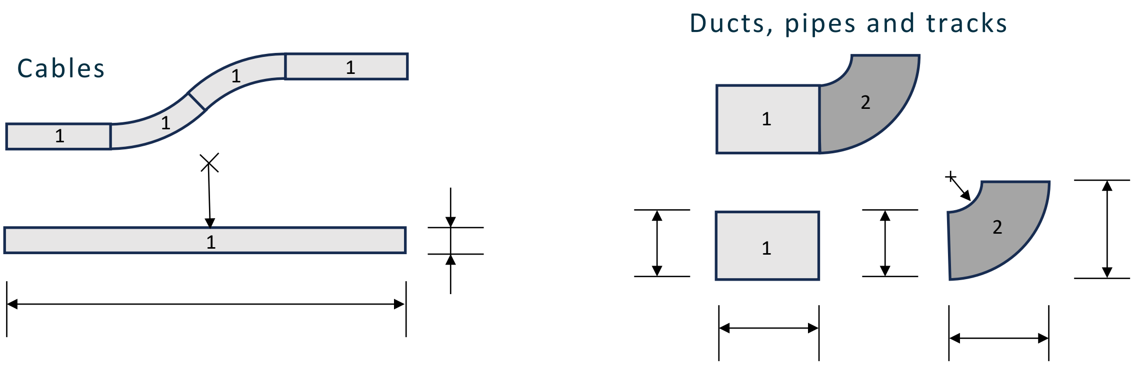


Figure 9 Linear and path-based shape examples

5.2.4. Assemblies

Some complex products consist of multiple functional components or modules grouped into one standardized object:

- **Assemblies:** Control panels, modular HVAC units, electrical cabinets and piping systems.
- **Complex shapes:** Products like pumps, air handling units, and boilers.

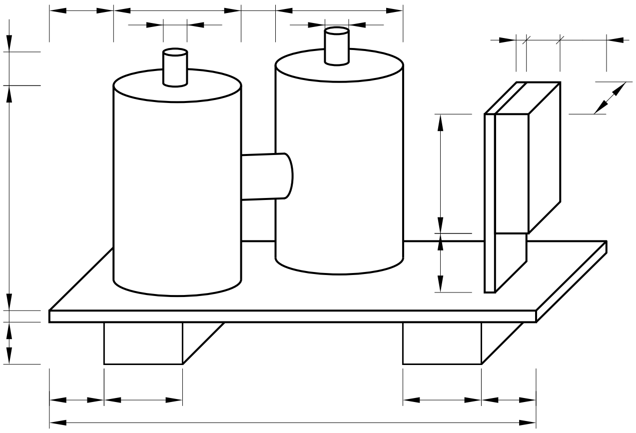


Figure 10 Example of a complex/assembly object with dimensions.

Key recommendations:
Provide critical dimensions for overall dimensions representing height, width and depth, as well as defining the necessary main components of the assembly with the same principles as Basic geometrical shapes.

5.3. Clearance spaces

Clearance spaces define the required three-dimensional zones around a product to ensure proper installation or operation, maintenance and safety. These spaces are critical for system designers and installers to for mounting, planning and early clash detection.

All clearance spaces in ETIM are represented as primitive rectangular boxes. For products that have irregular or complex clearance requirements, the space should be defined by the maximum extents of the required zone. A Modelling Class can define a maximum of two clearance spaces: Installation space and Functional space.

Installation space
The volume required to physically install, mount, and connect the product. It provides installers with the necessary clearance information and is used by planners for early-stage design coordination. The ETIM Installation space corresponds directly to the Installation Space defined in ISO 16757-2.

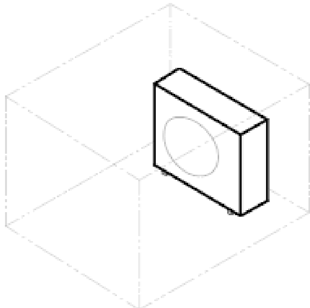


Figure 11 An outdoor unit for a heat-pump require a free clearance around it for the mounting process.

Functional space
The volume required for the product to be used, serviced, and operated safely after installation. It's essential for products with features like access panels, removable filters, door swings, or those requiring a minimum safety distance for heat dissipation. The ETIM Functional space is a pragmatic combination of two distinct spaces from ISO 16757-2: “Minimum Operation Space” and “Access Space”. The dimensions of the Functional space use the largest dimensions of these two including any safety clearances.

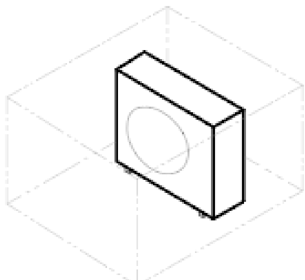


Figure 12 An outdoor unit for a heat-pump require a specific distance to its surrounding to ensure it can be used and maintained.

Dimensioning & offset rules
Spaces are dimensioned with three specific parameters and can be offset in three directions from the origin. The offset dimensions are placed at the shortest distance from the origin point to the space line. The space must be labelled in one of the views.

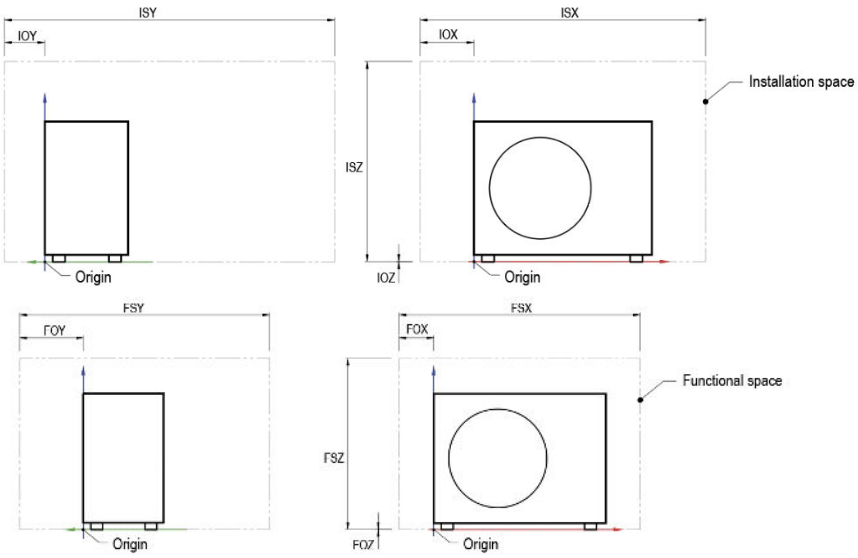


Figure 13 Space dimensioning with offset

Multiple spaces
An MC can at most have two spaces, one of each ETIM MC space type. Each one needs to have all three space and offset dimensions specified in the class.

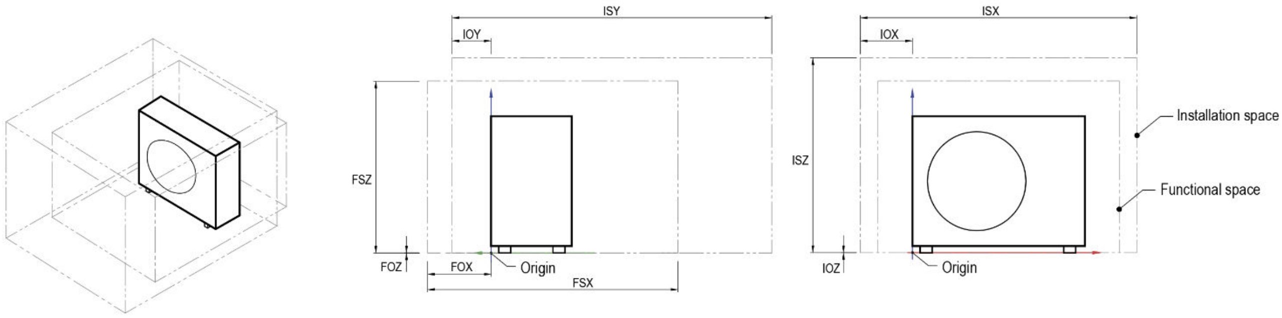


Figure 14 Installation and functional space used simultaneously.

5.4. Geometrical orientation and projection

5.4.1. Projection

ETIM MC employs the third-angle projection method, in which the object is conceptually placed behind the coordinate planes and projected orthogonally onto them. This projection style must be visually indicated using the third-angle projection symbol, as shown in *Figure 15*, and included in all parameter reference drawings.

Orthographic views are generated via parallel, orthogonal projections, resulting in flat 2D representations. While up to six standard views may be used, only those necessary to clearly convey the object's geometry should be included.

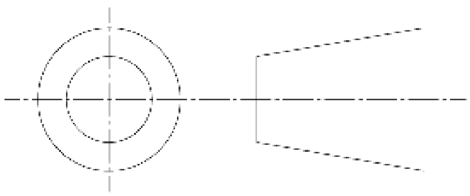


Figure 15 Projection symbol

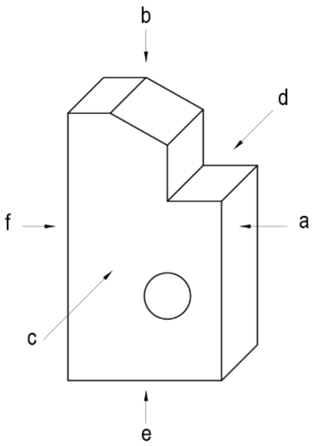


Figure 16 Perspective of views on a 3D object

Table 14 Standard orthographic views and descriptions.

View	Description
Front (a)	Displays height and width from the front perspective.
Top (b)	Displays width and depth from above.
Left (c)	Displays height and depth from the left.
Right (d)	Displays height and depth from the right.
Bottom (e)	Displays width and depth from underneath.
Rear (f)	Displays height and width from the rear.

5.4.2. Coordinate axes & origin

The coordinate system used in ETIM MC follows the right-hand rule for a three-dimensional Cartesian system. The origin, the intersection of the X, Y, and Z axes, is represented by a solid black dot in the drawing. Arrows must be placed at the end in the positive direction.

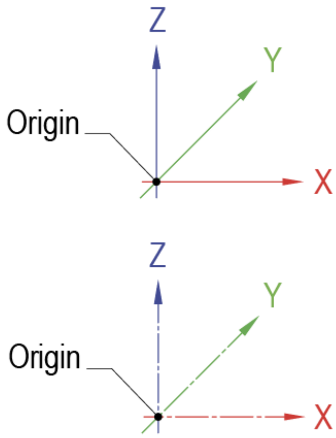


Figure 17 Coordinate axes and origin

X-axis: Positive direction points to the right

Y-axis: Positive direction points away from the viewer

Z-axis: Positive direction points upwards

Axis lines in drawings:

- Axis lines must appear behind the object lines and extend beyond the object boundaries.
- Endpoints of axis lines must include the axis letter in the positive direction.
- To improve readability, axis lines may be broken between the origin and their endpoints, if applied consistently in the drawing.

5.4.3. Origin placement

The point of origin should be positioned to reflect both the object's geometry and its intended functional context in a digital model. The following rules apply:

- **Asymmetrical objects**: Place the origin at the bottom-left corner of the front view for predictable alignment.
- **Symmetrical objects**: Centre the origin within the object's geometry to ensure balanced positioning.
- **Surface-mounted objects**: Set the origin where the object meets the mounting surface (e.g., wall or floor contact point).

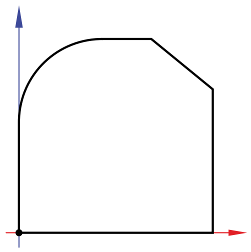


Figure 18 Origin placement on an asymmetrical object

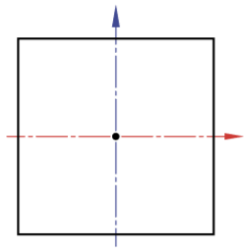


Figure 19 Origin placement on a symmetrical object

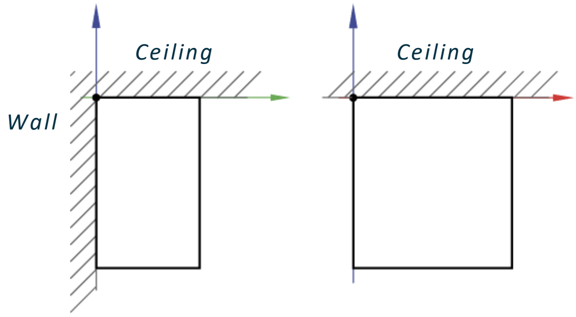


Figure 2018 Origin placement on a surface-mounted object

Connection-based objects:
For objects where geometry and alignment are defined primarily by connectors (e.g., fittings, valves, cables), the origin must be placed consistently to ensure interoperability and predictable behavior in digital modelling.

- **Single connector**
  - Place the origin at the center of the connector.
  - If the product has a more intuitive origin (e.g., a floor-standing tank), that position may override the connector-based default.
- **Multiple connectors**
  - Place the origin at the intersection of the connector centrelines.
  - This applies even if the connectors are arranged at angles or asymmetrically.
  - If the object is symmetrical, align the origin with the center of symmetry to ensure balanced placement.
  - For angled connectors, the intersection point may fall outside the object body—this is acceptable.

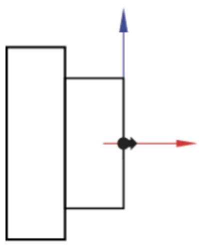


Figure 221 Origin placed at centre of the single connector.

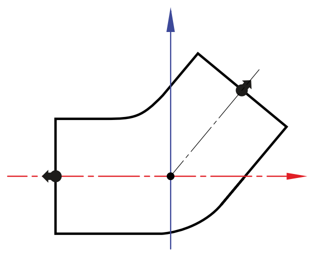


Figure 21 Origin placed at the intersection of connector centrelines.

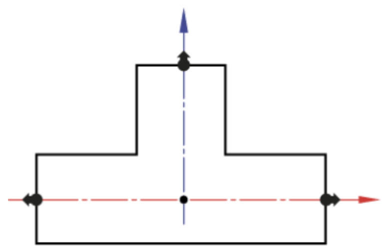


Figure 203 Origin placed at the centre point where all connector centrelines

Note: Always prioritize functional placement over geometric symmetry. The goal is predictable alignment within BIM systems.

5.5. Drawing views and layout

5.5.1. Positioning views

Drawings must include at least two views, with a minimum of three recommended for clarity. Avoid redundant views that add no additional information. Unnecessary details may be excluded to simplify and save drafting time, if clarity still is maintained. Drawings must be created at a scale ensuring readability. However, explicit scale notations or scale bars should be excluded from the drawing to maintain neutrality and scalability. For complex objects with many hidden parts, it is acceptable to reduce the number of hidden lines to improve clarity.

Use the standard view arrangement shown in *Figure 24*. View symbols are not required if the views are positioned accordingly. The views are derived from a 3D object, *Figure 25*, where each view, A-F, represents a projection of the object from different angles, providing a clear understanding of its geometry and features when interpreted together.

VIEW	VIEW FROM
A	The front
B	Above
C	The left
D	The right
E	Below
F	The rear

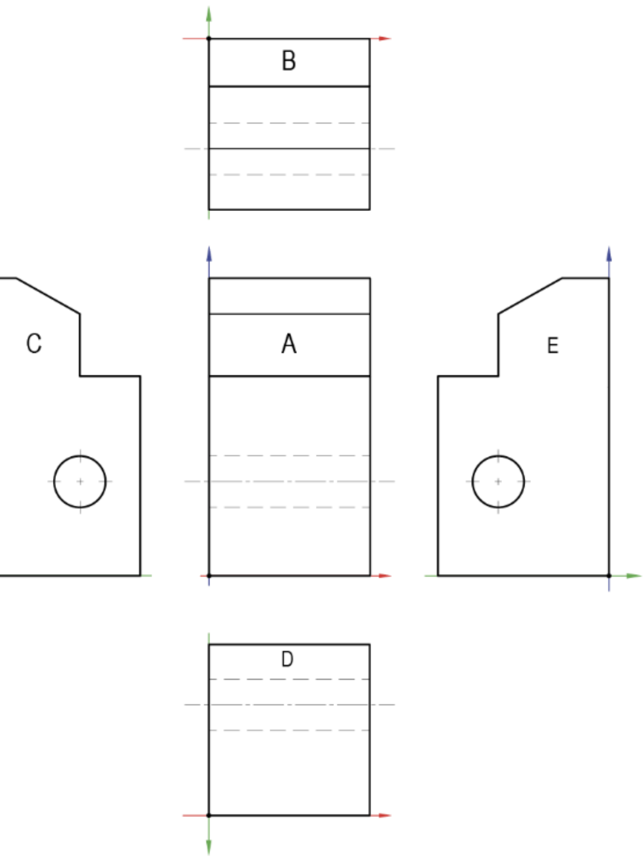


Figure 23 Placement positions of an object's views in a drawing and legend table

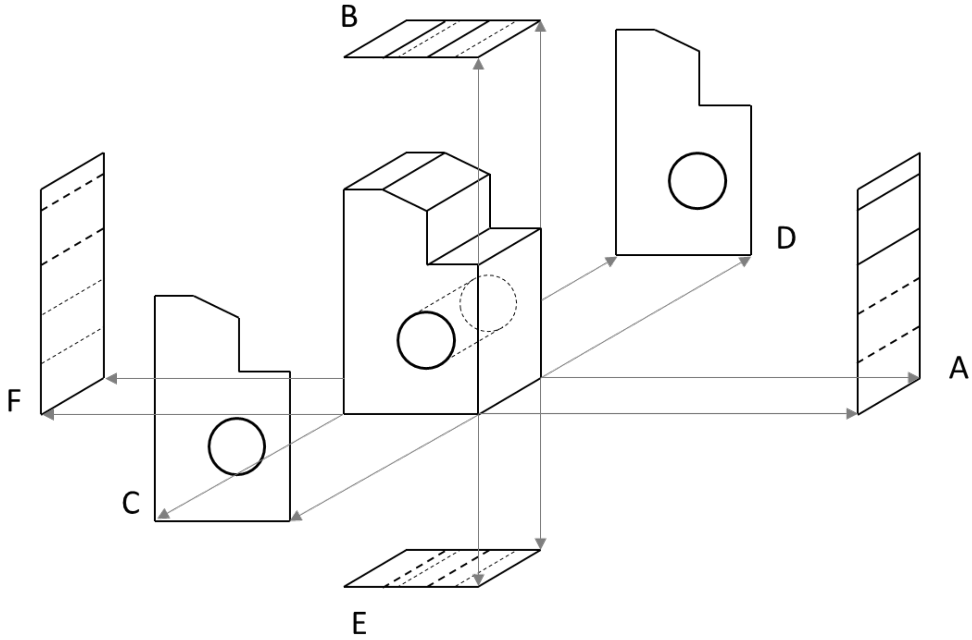


Figure 24 Views corresponding to a 3D object.

Note: *ETIM MC reference drawings intentionally omit scale annotations to ensure resolution independence and neutral display across systems. While this deviates from ISO 5455 in practice, it supports modern digital modelling use cases and interoperability through vector-based representations.*

5.5.2. View planes and specific views

A view is used to either enlarge a portion of the drawing, clarify alternate positions or parts. When positioning placed view planes, they shall be placed aligned with the view symbol as far as practical and not rotated. When placing a view without a view symbol, centerlines shall be used as reference instead.

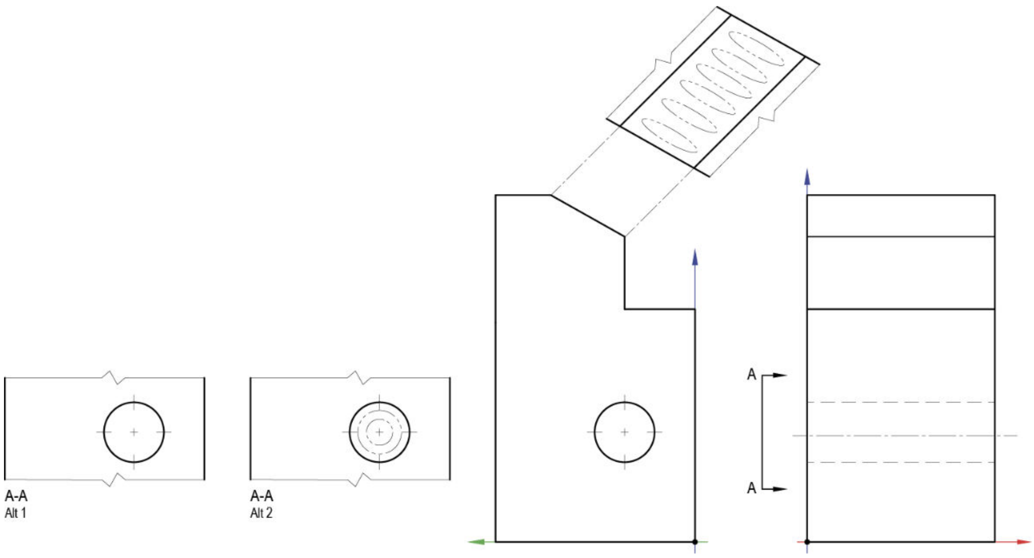


Figure 25 Placement of view planes and specific views.

5.5.3. Detail views

Detail views are used to enlarge specific features that cannot be clearly presented in the main projection due to scale or complexity. These views help clarify localized geometry without overcrowding the primary drawing.

Common uses include:

- Optional components (e.g., cut-outs, interface zones)
- Thread details or mounting features
- Symbols or PortCodes not readable at full scale

Guidelines for detail views:

- Label the magnified view clearly (e.g., Detail A)
- Indicate the area of focus in the main view with a circular dashed boundary
- Use alignment lines or leader arrows to connect the detail view to its origin point

Detail views are especially useful for showing variant options within a Modelling Class—such as optional displays, control interfaces, or mounting slots—without cluttering the main drawing.

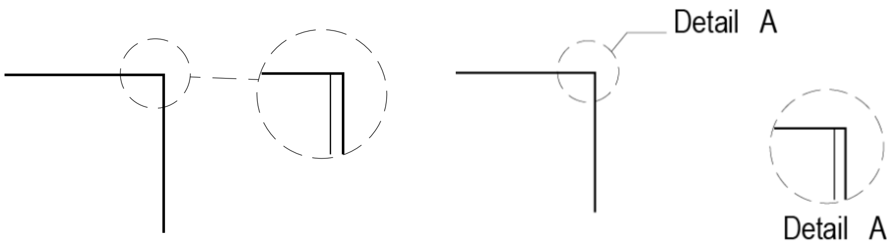


Figure 26 Examples of labelling details

5.6. Visual conventions and annotations

ETIM MC reference drawings utilize standardized line styles, symbols, and dimensioning rules to ensure consistency and clarity.

5.6.1. Line styles overview

Line styles in ETIM MC reference drawings follow standardized drafting conventions to ensure consistency, clarity, and compatibility. Line thickness should be adjusted based on the drawing scale and paper format. On smaller formats (e.g., A4), slightly thicker lines may improve readability, but care must be taken not to compromise detail visibility or reduce the overall visual precision.

Table 15 Standard line styles and attributes for ETIM MC drawings.

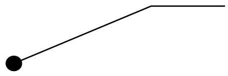
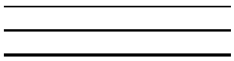
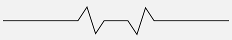
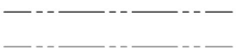
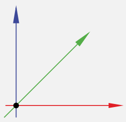
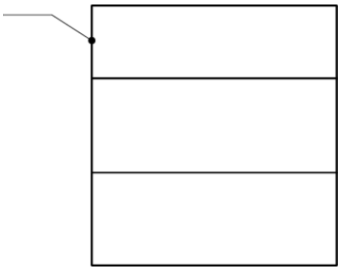
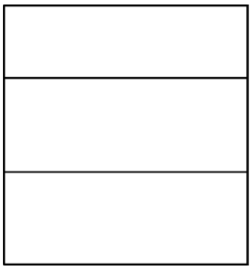
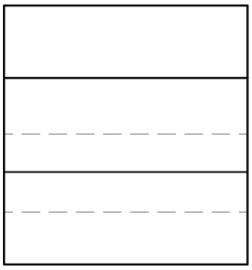
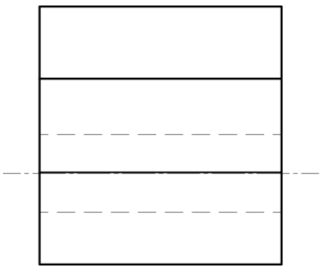
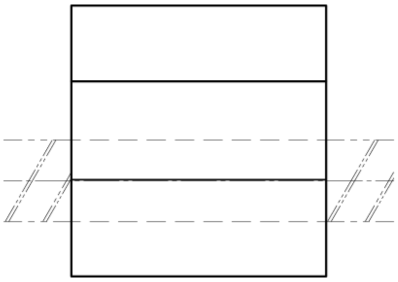
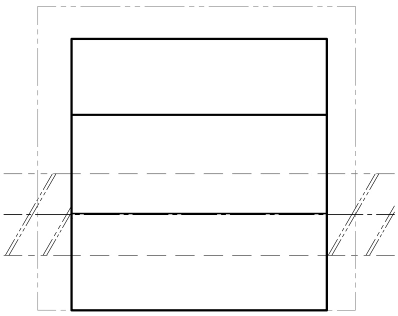
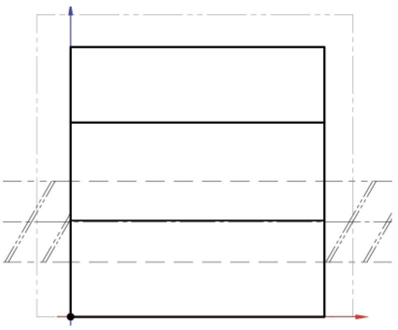
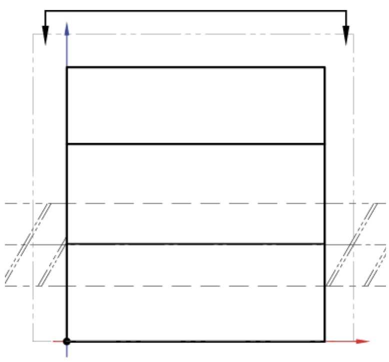
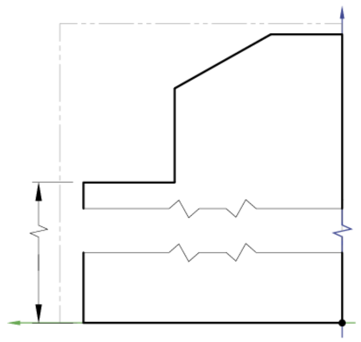
Feature	**Type**	**Thickness**		**Color**
		mm	pt	
Reference 	Solid	0,09	0,25	Black
Dimension 	Solid	0,09	0,25	Black
Object 	Solid	0,09 – 0,35	0,25 – 1,0	Black
Hidden 	Dashed line *Dash: 8 pt* *Gap: 4 pt*	0,18	0,50	Black
Center line 	Single chain *Dash: 16 pt* *Gap: 2 pt* *Dash: 2 pt* *Gap: 2 pt*	0,18	0,50	Black
Break 	Solid Zigzagged	0,18	0,50	Black
Phantom 	Double chain *Dash: 16 pt* *Gap: 2 pt* *Dash: 2 pt* *Gap: 2 pt* *Dash: 2 pt* *Gap: 2 pt*	0,18	0,50	Black Grey (128, 128, 128)
Axis line 	Solid/Single chain *Dash: 16 pt* *Gap: 2 pt* *Dash: 2 pt* *Gap: 2 pt*	0,18	0,50	Red (225,31,38) Green (83,173,71) Blue (61,73,153)
Section/viewing plane 	Solid	0,35	1,00	Black

Table 16 Descriptions and usage of standard line styles in reference drawings.

Illustrative examples	Line styles
	**Reference lines** Straight, thin solid lines used to connect elements of a drawing to dimensional codes or annotations. • Elbow segments should be used for angled lines. • Radius references do not require elbows.
	**Object lines** Depict visible, defining edges in the view. • Drawn with a solid black line. • Essential for defining shape and outer boundaries.
	**Hidden lines** Used to indicate features not visible in the current view. • Composed of short dashes. • Must start with a dash unless this would visually merge into a visible line. • Must connect cleanly at corners or tangent points on arcs. • Hidden lines must not be used for dimensioning.
	**Centre lines** Define symmetry, centre points of circles/arcs, and motion paths. • Composed of alternating long and short dashes. • Extend at least one-quarter beyond the object’s boundary. • At intersections, use a short dash. • Can also illustrate motion paths of rotating or sliding components.

	**Phantom – External geometry or optional parts** Used for alternate or repeated positions of parts or optional/hidden elements. • Uses black line colour. • Composed of one long dash followed by two short dashes. • Must start and end with a long dash.
	**Phantom – Spaces** Used for alternate or repeated positions of parts or optional/hidden elements. • Composed of one long dash followed by two short dashes. • Must start and end with a long dash. • Dimensions related to spaces must be placed externally using standard dimensioning. • Space outlines remain continuous even if the view is broken.
	**Axis lines** Used to show the X, Y, and Z axes in 3D space. • Follow centre line or solid line style. • Color-coded (Red for X, Green for Y, Blue for Z). • Start at the origin and extend in the positive direction, ending with the axis label.
	**Section/viewing plane** It indicates the perspective from a specific angle and placement. • A solid black line, same weight as object lines. • Ends with arrows pointing toward the surface being viewed.

	**Break lines** Used to represent an interruption in an object's length or shape when depicting shortened views. • Represented by a solid zigzag line. • Clearly indicate the excluded portion without ambiguity. • Object and dimension must both have the break indicated. • Spaces lines do not have any break indications.

5.6.2. Dimensions

Dimensions in ETIM MC reference drawings must ensure clarity, minimal redundancy, and structured layout. They are essential for accurately conveying the size, position, and shape of object features in 2D views.

Table 17 General rules for placing and styling dimensions.

Arrowheads	Dimension lines must terminate with arrowheads at both ends. Arrowheads should be placed inside the extension lines whenever possible. If space is limited, they may be placed outside. They must be 30° sloped and filled.
Line hierarchy	Dimension lines must not cross other dimension or extension lines. Extension lines may cross other lines if clarity is preserved.
Placement	Preferably place dimensions outside the object boundary. Internal placement is allowed only when it improves clarity, such as for internal features.
Extension line start gap	Leave a small visible gap (approx. 1.5 mm) between the object outline and the beginning of the extension line. Consistency in the drawing is prioritized over precise gap distance.
Rows	Position the first row of dimension lines at least 3x the text height away from the object (e.g., ≥ 6 mm for 2 mm text height). Space each additional row 2x the text height from the previous one.
View selection	Place dimensions in the view that best describes the true shape or size of the feature. If multiple aligned views are shown, prefer placing shared dimensions between views to reduce repetition.

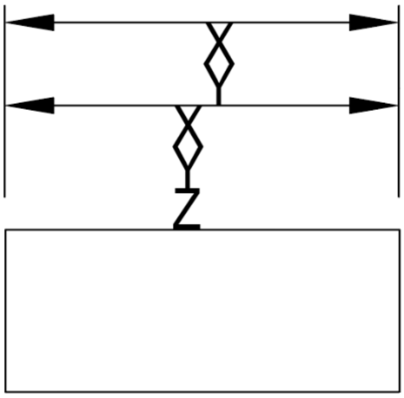


Figure 288 Dimensions, correctly spaced and with extension line gap

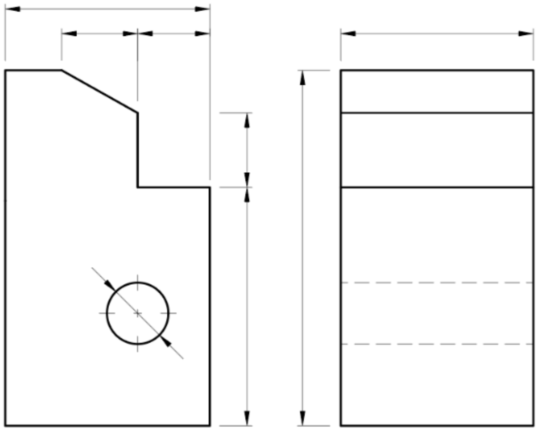
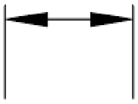
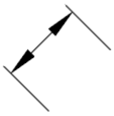
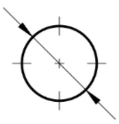
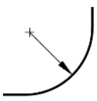


Figure 279 Correctly placed dimensions in views and dimension groups

Dimension types

The following types of dimensions are used to define distances, diameters, radii, and angles in technical drawings. Each type should be applied based on the geometry being measured and the view that best represents the feature.

Table 18 Types of dimensions and their definitions.

Illustration	Type	Definition
	Horizontal	Measures width or length left-to-right.
	Vertical	Measures height bottom-to-top.
	Angled	Measures sloped distances not aligned with the main axes.
	Diameter	Shows the full width of a circular feature and should be presented in a rectangular or side view.
	Radius	For arcs less than 180°, the arrow points from the arc centre to its perimeter.
	Angular	Measures the angle between two intersecting or referenced lines.

Rounding and cylindrical features

- **Rounded corners or fillets**
  Rounded corners or fillets should be dimensioned using radius values. Extension lines may reference the theoretical sharp intersection for clarity.

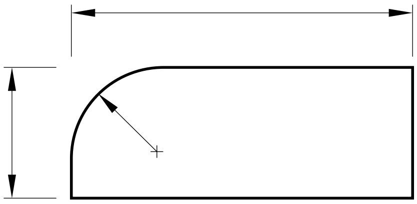


Figure 300 Dimensioning rounded corners at theoretical intersection and with radii.

- **Cylinders and holes**
  Diameter and length must be dimensioned in rectangular views. For holes, dimensions must be placed in circular views with location given from the centre point.

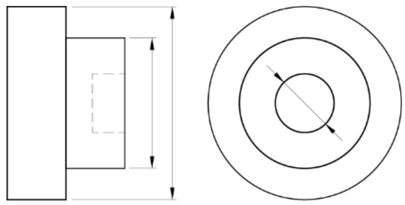


Figure 291 Dimensioning cylinders in rectangular view and holes in circular views

Dimension organization

- **Dimension chain**
  A sequence of consecutive dimensions along an axis, sharing a common reference.

Figure 31 Linked dimensions in a chain

- **Dimension group**
  Multiple dimensions with differing orientations. Arrange by size (largest to smallest) or follow logical reading order.

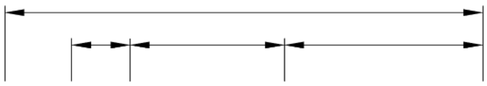


Figure 32 Grouped dimensions

- **Short dimensions**

  When a chain dimension is too short to fit arrows on both sides, use an oblique stroke (a diagonal slash) between the extension lines instead of arrows. This symbol indicates a dimension is present, even when space is limited.

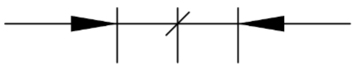


Figure 33 Tight spaced dimensions using a diagonal line instead of arrows for clarity

- **Overall size**
  Always include a total dimension to represent the full extent of an object or sub-feature, unless already implied through dependent geometry.

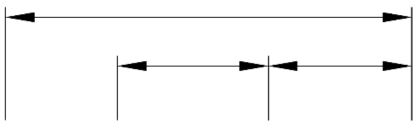


Figure 34 Grouped dimensions with an overall dimension, leaving out the need of a third dimension in the chain.

- **Avoid redundancy**
  Do not repeat dimensions unless necessary for clarity. When using chain dimensioning, leave out one segment to avoid repeating the same total.

5.6.3. Hatching

Hatching is a pattern of evenly spaced lines used to indicate that part of the modelled object is recessed into, penetrates, or is fixed to a host surface or material. It is optional, but should be used when it helps clarify the installation context.

Table 19 Hatching requirements for installation context in drawings.

Item	Requirement
Pattern	45 ° parallel lines Line weight: 0.18 mm/50 pt Spacing: 2 mm/500pt Color: Grey (102,102,102)
Boundary	Hatch must stop precisely at the host surface line, do not cross into visible object edges.
Annotation (optional)	If multiple host materials appear in one drawing, add a brief note (e.g., Ground, Wall, Ceiling) outside the hatch.
Clarity	Hatch only the minimum area needed, avoid cluttering the drawing.

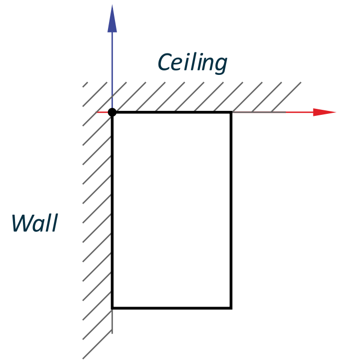


Figure 35 Hatched area indicating the corner of the ceiling and a wall.

5.6.4. Lettering and numerals

All letters in the drawing shall be capital letters, with exceptions for certain drawing codes, and numbers shall be Arabic numerals. The letters and numerals shall be annotated in a horizontal reading position as far as it is practical and cannot overlap with each other. Models should have as little language dependent indicators as possible, always in English and avoid abbreviations.

Table 20 Font styles and text sizes for reference drawings.

Drawing feature	Font type	A4 Text height		A3 Text height	
Drawing title, Class code	**Arial Narrow Bold**	20 pt	5,0 mm	24 pt	6,0 mm
Views, table titles, labels	Arial Narrow	10 pt	2,5 mm	14 pt	3,5 mm
Drawing codes, tables, text	Arial Narrow	8 pt	2,0 mm	10 pt	2,5 mm

5.7. Drawing codes and optional features

5.7.1. Drawing codes

Drawing codes should consist of a letter followed by a number.
The recommended assignment of numbers depends on the dimensioning method but always starts from the origin point:

- **Chain dimensioning**: When dimensions form a chain of measurements along an axis, the dimensions shall be numbered sequentially starting from 1.
- **Group dimensioning**: When multiple dimensions represent portions of a larger overall dimension, the largest dimension shall receive the lowest number, as far as practical.

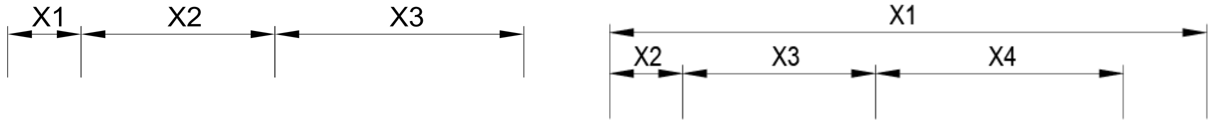


Figure 36 Examples of assigning drawing codes to dimensions with chain and group dimensioning

Table 21 Table of drawing codes in ETIM MC. (n=number)

Drawing code	Dimension/Type	Axis/Description	Letter case
X*n*	Distance parallel to axis	X-axis	Capital
Y*n*	Distance parallel to axis	Y-axis	Capital
Z*n*	Distance parallel to axis	Z-axis	Capital
XY*n*/XZ*n*/YZ*n*	Distance in a diagonal or angled direction	Non-axial (inclined or slanted)	Capital
ISX/ISY/ISZ	Installation space distance parallel to axis	X, Y, Z	Capital
FSX/FSY/FSZ	Functional space distance parallel to axis	X, Y, Z	Capital
IOX/IOY/IOZ	Installation space offset distance parallel to axis	X, Y, Z	Capital
FOX/FOY/FOZ	Functional space offset distance parallel to axis	X, Y, Z	Capital
A*n*	Angle	Non-axial (angular features)	Capital
D*n*	Diameter	Non-axial (circular features)	Capital
R*n*	Radius	Non-axial (e.g., bends, fillets, roundings)	Capital
S*n*	Thickness	X, Y, Z or non-axial	Capital
CA	Coordinate position	Coordinates of a specific point	Capital
DV	Directional vector	Direction of a specific vector	Capital
CT*n*	Connection Type	-	Capital

5.7.2. Optional features and drawing codes

Optional features are geometric elements of a Modelling Class that are commonly found in real-world products but are not required to define the object. These may include displays, cutouts, extensions, or mounting options. Their inclusion is determined by the specific product variant or use case.

Stakeholder perspectives

- **For manufacturers**
  Optional features are not mandatory to populate in product data. If a product lacks the feature, the corresponding dimension may be omitted or assigned a value of zero.
- **For designers and modellers**
  Optional features represent geometric components that may be toggled on or off during object generation. This allows flexible configuration of BIM or CAD objects to match product variants.

Representation in drawings

- Optional features must be indicated in reference drawings by enclosing the drawing code in parentheses, e.g., (X1).

- These features should be clearly distinguishable from mandatory ones, using consistent visual notation.

Representation in the Feature table

- Drawing codes must be listed without parentheses in the feature data table.
- Dimensional values for optional features can be left undefined or set to zero if not applicable to a product instance.

General drawing Note

If any optional features are present, the following general note must be included in the drawing:

NOTE: *"Drawing codes in parentheses indicate optional features that are not required in every product implementation."*

Example

- **Feature table**: X1
- **Drawing code**: (X1)
- **Drawing note**: as above

Implementation recommendation

Object creators and tool developers are encouraged to implement a toggle mechanism for optional features in generated objects. This supports flexible configuration of Modelling Classes to match different product variants, while preserving consistency in geometry and parametric structure.

6. Connection modelling

Accurate and standardized connection modelling is a cornerstone of the ETIM Modelling Class (MC) framework. In many technical domains, components must not only be represented geometrically, but also semantically linked through connection points that define how they integrate within broader systems.

To ensure consistency, reusability, and system-level interoperability, ETIM MC employs a dual-component connection model built on:

- **PortCodes**
  Unique numeric identifiers that define and distinguish individual connection points on an object.
- **Connection Type Classes (CTs)**
  Reusable class entities that standardize the geometry, dimensions, and semantics of common connection interfaces (e.g., threaded joints, sockets, flanges).

This modular approach decouples connection logic from product geometry, allowing designers and developers to build scalable, configurable models that remain compliant across digital platforms and software environments while allowing for manufacturers to clearly define what type of connection it is. PortCodes can be used independently from

Connection Types being used to indicate connection points, but Connection Types must use PortCodes. PortCodes indicates that it can be connected to another part in the system. Compared to Connection Types, these connection points are typically not universally standardised interfaces.

6.1. PortCodes

PortCodes are numeric identifiers that uniquely define each connection point on a modelled object. They link connection geometry to dimensional data and support flow, direction, and placement definitions in both 2D drawings and parametric features.

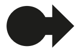


Figure 37 PortCode symbol

Table 22 PortCode numbering rules

Type	Description
PortCode 0	Default, used when a feature is not linked to a specific connection point.
PortCode 1	Typically, the primary inbound or most prominent connection. If unclear, use the bottom-left connector in the front view.
PortCode 2..n	Assigned sequentially in a clockwise direction, starting from Port 1. • **Exception for branching connections (e.g., T-joints):** Assign Port 2 to the connector directly opposite Port 1 (the main outlet), and assign branch ports (any side-facing connectors) as PortCodes 3, 4, etc. • **Clockwise assignment view priority:** Front, Left, Right, Back, Top, Bottom.
Multiple identical connections	Use a range notation (e.g., [1–n]) in reference drawings for repeating connection patterns. The range upper limit must have a limit and be within the amount of specified PortCodes.

6.2. Connection Type Classes

Connection Type Classes (CTs) define standardized connection interfaces that can be reused across multiple Modelling Classes (MCs). Each CT captures only the interface-specific characteristics, such as diameter, thread type, or connector form, without duplicating core object geometry. The current scope of CTs is for standardised mechanical connections within the HVAC and plumbing industry.

Benefits of CTs

- **Standardization**: Defined once, reused in many classes.
- **Operability**: Ensures consistent handling across software and disciplines.
- **Simplified modelling**: CT placeholders reduce geometric complexity and support modularity.

Key principles

- CTs are isolated from MCs and treated as attachable modules.

- CTs are version-controlled independently.
- CTs must not duplicate MC geometry for the same PortCode.
- If no CT fits, a new one must be created through the RFC process.
- When multiple CTs apply to one PortCode, all valid types should be linked for downstream selection.

Important: CTs define the interfaces geometry only. The MC still controls the objects core geometry and critical dimensions.

CT drawing format

- **File format**: SVG
- **Naming:** *[ConnectionTypeClasscodeID]_V[VersionNumber].svg*
  Example: CT000001_V4.svg
- **Compliance:** Must use the latest ETIM drawing template and line style rules

6.2.1. Visual representation of CTs in drawings

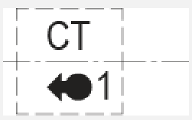
In MC reference drawings, CTs are not drawn to scale or detail, but symbolically represented to show:

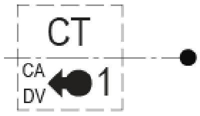
- The type of connection interface linked via the CT,
- Its spatial location, and optionally,
- Its direction and position using coordinates or vectors.

Drawing symbols
The CT placeholder symbol must be composed as follows:

- Dashed rectangle surrounding the connection placeholder.
- Label "CT" centred at the top of the box.
- PortCode shown as a solid black circle (representing the port) with the code (e.g., 1) to its right.
- CA and DV placement:
    - CA (Coordinates): Above the connector symbol.
    - DV (Directional Vector): Below the connector symbol.
    - Both labels are aligned vertically on the left side of the symbol.

Table 23 Symbols used for Connection Types in ETIM Modelling Classes

Symbol	**Description**
	Connection Type assigned with PortCode 1 without designated origin. Shares the origin with the object in all three axes, see *Figure 39.*
	Connection Type assigned with PortCode 1 with designated origin. It has a designated origin point indicating where the connection is located, see *Figure 40*.

	Connection Type assigned with PortCode 1 with designated origin and specific coordinates. It has a designated origin point and its exact position defined by coordinates.
	Connection Type assigned with PortCode 1 with designated origin, specific coordinates and a directional vector. It has a designated origin point, its exact position defined by coordinates and its direction specified with a vector. See *Figure 41.*

These symbols must be used consistently to indicate how the CT integrates with the MC in a 3D space. The symbols should be placed in all views where relevant to increase clarity.

Example: CTs using object origin

In *Figure 39*, both Connection Types (CT1 and CT2) share the same origin point as the MC object. This means that each CT is aligned with the objects three principal axes. Individual Coordinates (CA) or Directional Vectors (DV) can be used when using the objects origin point.

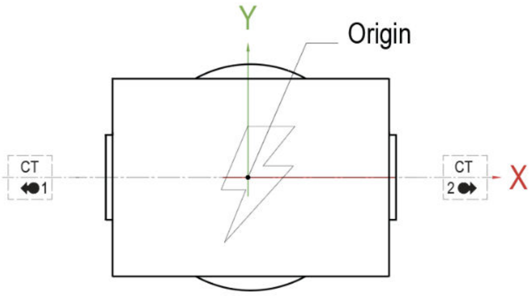


Figure 38 Connection Types assigned to PortCode 1 and PortCode 2, without designated origin.

Example: CTs using individual origins

In *Figure 40*, neither of the Connection types can be positioned using the objects origin. Therefore, each CT has its own individual origin point. In this example, the locations of the CTs are specified by dimensional parameters and centrelines, and no Coordinates or Directional Vectors are needed.

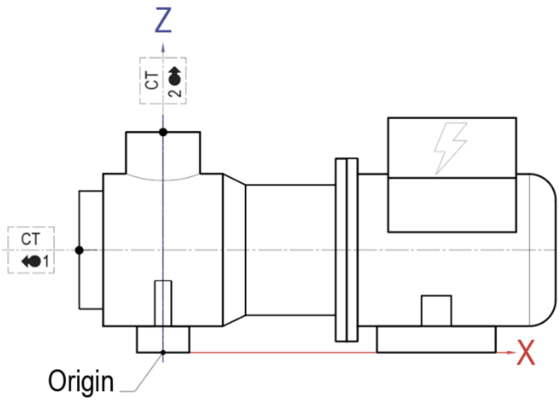


Figure 39 Connection Type assigned with PortCode 1 and PortCode 2, with designated origin placed on the object.

Example: CTs positioned with CAs and DVs

For objects where the connections require precise positioning, direction and can have many possible positions. The CTs in *Figure 41* are given precise positions by Coordinates and the direction with Directional Vectors while being represented in the drawing at a common position.

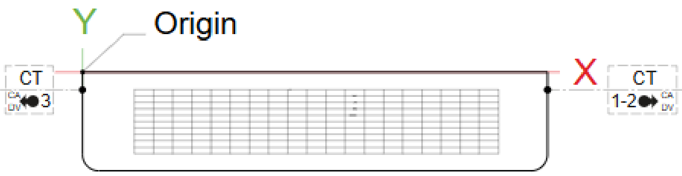


Figure 40 Connection Type assigned with PortCode 1 with designated origin, specific coordinates and a directional vector.

6.2.2. Reference drawing requirements for CTs

CTs must follow the same visual style and SVG-based formatting as MC drawings, but with a different emphasis:

Key differences from MC drawings

- **No absolute origin**: CTs do not define global coordinate positions.
- **Neutral geometry**: Only dimensions relevant to the connection interface.
- **Reference-only axis system**: Two dashed lines intersecting at the origin to indicate orientation, not labelled X/Y/Z.

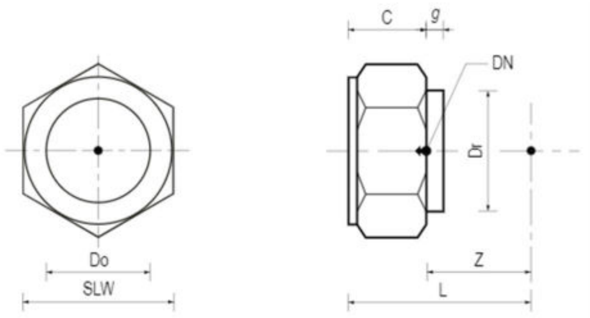


Figure 41 Connection type class with dimensions and the origin dot as the insertion point.

Drawing requirements

- A visual origin dot at the connection point
- Two dashed lines intersecting at 90°, centred at the origin
- If dimensional details (e.g., thread depth, diameter) are needed:
  - Use chained dimensions from or to the axis lines
  - Ensure consistency with MC drawing conventions

Important: CT drawings must never override object-level geometry, but provide reusable, modular connection logic.

6.3. Implementation rules and best practices

PortCodes and Connection Type Classes (CTs) must be applied with care to preserve clarity, modularity, and consistency in ETIM Modelling Classes. The following rules and techniques support accurate linking, avoid data duplication, and ensure scalable use of reusable connection components.

6.3.1. Linking CTs to Modelling Classes

PortCodes serve as the bridge between a Modelling Class and its Connection Type Classes. Each PortCode must point to a specific CT, which defines the relevant connection interface (e.g., thread, socket, flange).

- Use PortCodes to isolate interface definitions from core object geometry.
- Treat CTs as external modules—only responsible for the connection, not the object shape.
- Ensure the MC still defines its full geometry independently (e.g., length, width, mounting height).

Illustrative example
The following drawing shows a simple two-port radiator stop valve. Port 1 and Port 2 are assigned distinct CTs using consistent symbols and placement rules. The CTs define only the connection interface (e.g., thread type), while the MC defines the core object dimensions and geometry.

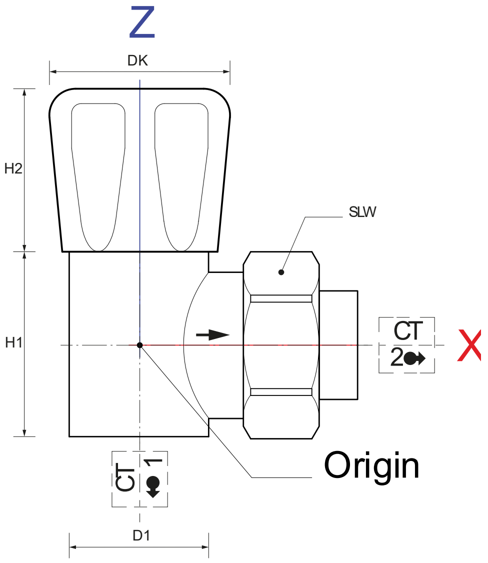


Figure 43 Radiator stop valve with two connections symbolized with connection type symbols. The connections share the origin of the object.

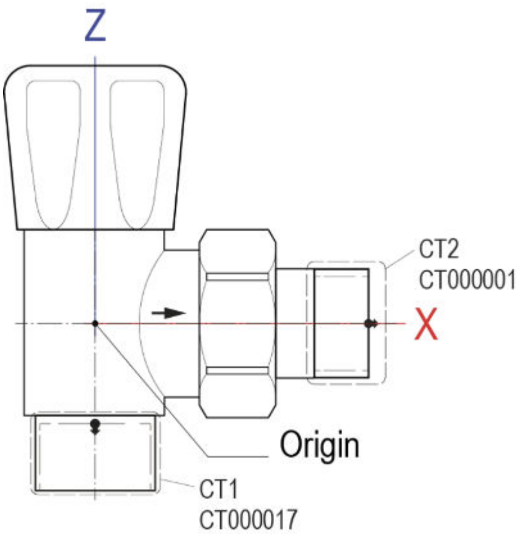


Figure 424 Conceptual view of the radiator foot valve and the connection types.

Table 24 Connection Types assigned to ports

Code	Version	Port	Description
CT000001	4	1	Pipe end
CT000001	4	2	Pipe end
CT000017	4	1	Pipe sleeve

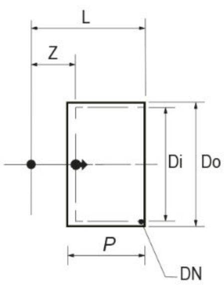


Figure 45 CT000017

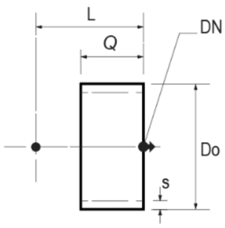


Figure 44 CT000001

6.3.2. Avoiding redundancy

To prevent conflict or ambiguity:

- Never duplicate dimensions between an MC and a CT.
- Avoid defining the same feature (e.g., outer diameter) in both MC and CT.
- CTs are a supporting geometry, placed within the MC.

CTs should only include features specific to the connection interface, which are inherited into the product-instance of the model when the CT is selected. All object-level dimensions and non-interface features belong to the MC.

CTs are intended for reuse across classes and must remain stable over time:

ETIM International

- Do not create variants of CTs unless necessary.
- Small positional differences (e.g., rotated placement or offset) should be handled within the MC using coordinates or vectors—not by duplicating CTs.
- Link multiple valid CTs to a single PortCode if multiple CTs are supported (e.g., threaded and flanged ends).

6.4. Coordinates & Directional vectors

To enhance modelling efficiency, ETIM MC supports the use of Coordinates (CA) and Directional vectors (DV) to define the position and orientation of connection points. These tools allow for precise and scalable positioning of Connection Type Classes, particularly when connectors are asymmetrically placed or aligned at non-standard angles.

CTs using coordinates and/or vectors must have their placeholders explicitly marked with corresponding CA and/or DV feature codes in the Modelling Class.

6.4.1. Coordinates

Coordinates (CA) define the exact spatial position of a connection point relative to the origin of the Modelling Class. They are essential when a connection is:

- Not centred on the origin,
- Asymmetrically arranged, or
- Difficult to define using dimensional parameters alone.

Coordinates are defined as feature type C in the feature list and expressed as a signed triplet of decimal values in millimetres, aligned with the ETIM MC global coordinate system:

- X-axis: Negative (left) to Positive (right)
- Y-axis: Negative (back) to Positive (front)
- Z-axis: Negative (down) to Positive (up)

Example:
CA1(+100, +50, -30)
This defines the location of Connection 1 as 100 mm to the right, 50 mm forward, and 30 mm below the Modelling Class origin.

Drawing code convention
Each coordinate feature must be assigned a unique drawing code using the format CA*n*, where:

- CA indicates a coordinate assignment
- *n* is the corresponding PortCode or feature number
- Example:
  CA1 refers to the coordinate feature assigned to PortCode 1.

Coordinate assignment rules

- Coordinates must be defined when a connection point does not align with the origin.
- If a connection shares the origin, it must be explicitly recorded as (0, 0, 0) for clarity.
- All coordinate features must be included in the Modelling Class feature table, with feature type C and appropriate units.
- Coordinates are interpreted in the right-handed Cartesian system defined in section 5.4.2.
- Coordinates must only be used for positioning connections. Their use for non-connection features is not permitted.

6.4.2. Directional vectors

Directional vectors (DV) define the orientation of a connection point within the Modelling Class coordinate system. They are particularly useful for modelling angled or variable-facing connections and ensure parametric consistency across brand variations without the need for multiple geometric models.

Example:
DV1(0, 0, +1)
This indicates that Connection 1 is oriented vertically upwards in the positive Z-direction, such as a pipe that connects at the top of the object.

Vectors must be defined:

- When the connection is angled or rotated from standard axis alignment.
- When orientation varies across similar product variants or brand series.

Drawing representation
Directional vectors must be illustrated in the dimensional drawing using:

- A cross symbol at the vector origin.
- A directional arrowhead (no shaft) showing the vector direction.
- A drawing code, placed adjacent to the vector

Figure 46 Vector symbol

Drawing code convention
The default drawing code format is DV*n*, where:

- DV stands for Directional Vector
- *n* is the corresponding PortCode or feature number

Directional vectors must be defined in the feature table with feature type C, using signed decimal values to express direction along the X, Y, and Z axes.

Example:
DV1 refers to the vector direction of PortCode 1.

Table 25 Common direction vectors

Vector	Descriptive direction
[1, 0, 0]	Positive X direction (towards the right).
[-1, 0, 0]	Negative X direction (towards the left).
[0, 1, 0]	Positive Y direction (towards the front).
[0, -1, 0]	Negative Y direction (towards the back).
[0, 0, 1]	Positive Z direction (towards the top).
[0, 0, -1]	Negative Z direction (towards the bottom).
[0.5, 1, -1]	Vector angled towards the front, slightly left, and downward.

6.5. Flow directions

Flow direction must be clearly indicated for all objects where internal or external media flow is relevant to the object's function.

Single flow
For unidirectional systems, place a straight arrow near the entry point to indicate the intended direction of flow.

Multiple flows
For objects supporting multiple, distinct flows (e.g., supply and return lines), each flow must be identified by extending the PortCode with a flow number (e.g., 1-1, 3-1). In such cases, place a straight arrow near the exit point of each flow path to clarify direction.

These arrows are symbolic and should follow standard line styling for annotations. Their consistent use ensures interoperability and correct interpretation across platforms and disciplines.

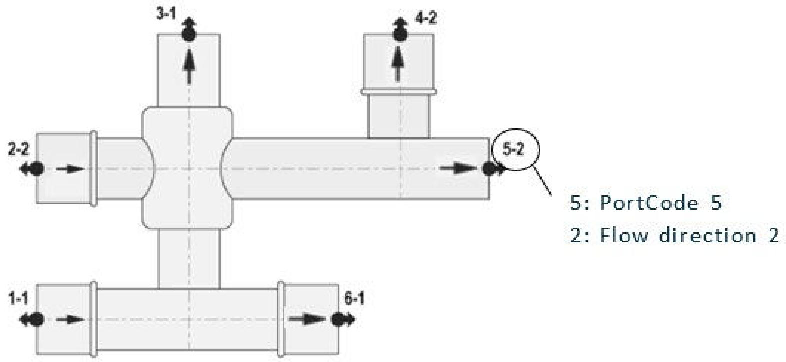


Figure 47 Example of multiple flow directions indicated by PortCodes and Flow Numbers

6.6. Working lengths and connection features for insert-based connectors

Many products across technical domains, use insert-based connection mechanisms, commonly referred to as male/female connectors. This section standardizes how such interfaces should be represented in Modelling Classes.

Working length, connector and dimensional guidelines:

- **Female connections** (e.g., sockets, sleeves, internal threads)
  Must include a working length when this depth is relevant to determine the total installed system length. The working length represents the insertion depth of the connected counterpart.
- **Male connections** (e.g., spigots, external threads)
  Are generally modelled without additional working length. The extension of the male connector defines the spatial boundary.
- **Linear elements** (e.g., tubes, ducts, conduits)
  Are defined by their full installed length, without separate working lengths. Their geometric representation is equal to the actual span between connectors.
- If a threaded or insert-based connector affects installation geometry or internal flow characteristics, the depth must be dimensioned explicitly.

  Note: These working length standards are critical for systems requiring accurate total length calculations, particularly in hydraulic and ducted installations.

Example: An internally threaded connection must show its thread depth if this limits the effective flow or determines the insertion boundary of a mating component.

Figure 48 Sleeve and pipe connections: separate (top) and assembled state (bottom).

6.7. Nominal sizes in connection modelling

Nominal sizes, such as Nominal Diameter, Bore, or Pipe Size, are widely used in industry to label and classify connection interfaces across systems. These values act as standardized identifiers to ensure compatibility between components but do not represent actual physical dimensions. In the ETIM Modelling Classification standard, nominal sizes are not used to define geometry. Instead, all geometrical features in the model should be derived from measurable parametric features, such as:

- **Inner diameter**
- **Outer diameter**
- **Wall thickness**

For threaded connections, the geometry is defined using the pipe-end dimensions, ensuring consistency across model implementations while relying on nominal labelling. When the nominal size is relevant for classification purposes, it should be represented as a feature label with alphanumeric values, following standard nominal size designations.

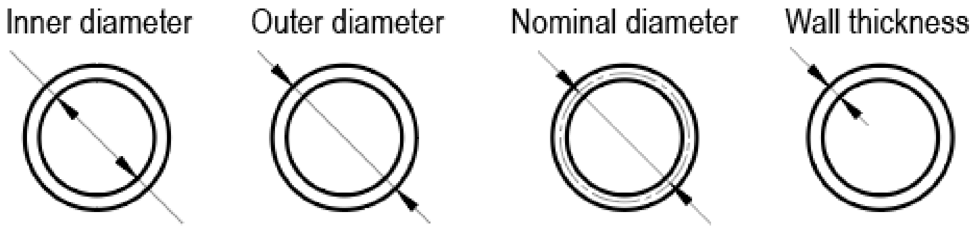


Figure 49 Diameter measurements for circular objects

6.8. Controls, actuators and components

Multiple controls (handle, motor etc.) can be part of one model. In that case the controls are numbered. The numbering should refer to a legend, explaining the meaning of the number, referring to an ETIM value code.

When multiple controls are attached to an object, each must be linked through a separate PortCode.

Controls must be numbered, referenced in a legend in the drawing (see *Figure 48*) and follow the same order and names as in the feature table.

	Operation	EV Code
1	Electric motor	EV020343
2	Handle	EV020000
3	Handwheel	EV020003

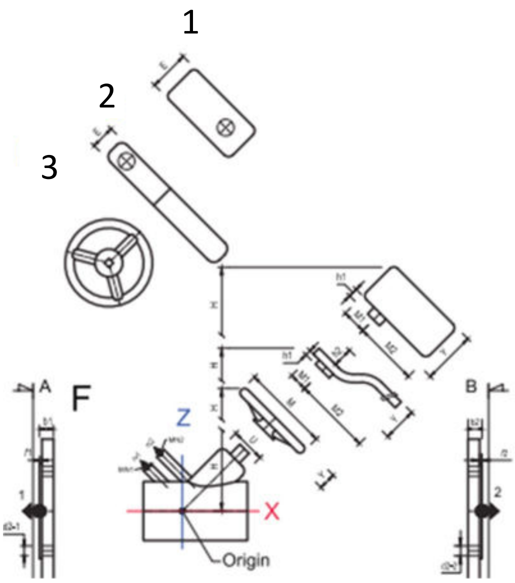


Figure 50 Example drawing of an object with multiple controls, including a reference legend of the controls

6.9. Electrical connections

Electrical connections are used to indicate the position of the connection points on products where the electrical connection is a part of the installation of the product. They should only be included when the product requires a fixed connection (e.g., heat pump) and not when cables commonly are pre-installed or have a standardized connection for cables (e.g., routers, display units). The electrical connections in ETIM MC do not use Connection Type Classes and are defined by PortCode, Coordinate, Feature and drawing code with a symbol in the 2D drawing for annotation. The purpose is to enable BIM software, engineers, installers and manufacturers to localize the connection point on the product. Common types of allowed electrical connections are:

- Power
- Thermostatic
- Earthing

Other electrical connections might be acceptable if motivated as important for the installation or for the users of the object.

Non-accepted electrical connections

Plug-and-play connections can only be added for visual recognition and are not allowed as a Connection Type nor should be added as a connection to the MC to avoid complexity.

Examples of plug-and-play connections:

- USB (USB A/B/C)
- HDMI, DisplayPort
- Ethernet
- Audio jacks (2.5 mm, 3.5 mm, RCA)

7. Index of appendices

The appendices listed below form an essential component of these guidelines and will be maintained as living documents, subject to ongoing updates:

- **Appendix 1:** Guidelines glossary
- **Appendix 2:** Drawing compliance checklist

8. References and standard alignments

This chapter outlines the reference materials, licensing information, and external standards that have influenced the ETIM Modelling Class framework. While certain standards have influenced the development of ETIM MC, formal compliance or certification is not implied unless explicitly stated.

Internal references
ETIM International Classification Guidelines, ETIM International, latest version.
Available at: https://www.etim-international.com/downloads

License information
Open Data Commons Attribution License (ODC-By) v1.0, Open Knowledge Foundation.
Available at: http://opendatacommons.org/licenses/by/1.0

Referenced external standards and alignment
The ETIM Modelling Class (MC) guidelines are developed in alignment with key international standards. The following table provides a detailed statement of alignment for each referenced standard. This statement is a self-declaration by ETIM International and not a third-party certification. These alignment claims apply to all ETIM MCs created or revised under this version of the guidelines, effective from the documents release date.

- **Aligned:** The principles and outputs of the ETIM MC guidelines meet the requirements of the referenced standard.
- **Partially aligned:** The guidelines adopt specific principles or requirements from the standard, but do not implement the referenced standards full scope.

Table 26 External standards referenced and their compliance status.

Standard (title)	Alignment	Note
ISO 128:2020 *Technical product documentation - General principles of representation*	Aligned	ETIM MC reference drawings follow layout and projection standards consistent with ISO 128.
ISO 129-1:2018 *Technical product documentation - Dimensioning*	Aligned	Dimensional annotations follow ISO 129 conventions for line placement, clarity and geometry reference.
ISO 3098-1:2015 *Technical product documentation - Lettering*	Aligned	Font styles, cases and sizes in ETIM MC drawings conform to ISO 3098 specifications.
ISO 5456-2:1996 *Technical drawings - Projection method*	Aligned	Every drawing employs the third-angle orthographic projection method: the standard symbol is included; views are arranged per the orthographic view table; and only views needed for clarity are shown.

ISO 12006-3:2021 *Object-oriented classification framework*	Aligned	ETIM MCs are structured using object-oriented principles with universal identifiers and hierarchical modelling consistent with ISO 12006-3.
ISO 19650-1:2018 *Information management using BIM - Concepts and principles*	Aligned	ETIM MC supports structured data modelling, LOD 200 geometric templates and versioning aligned with information-management requirements in the design phase.
ISO 19650-2:2018 *Information management using BIM - Delivery phase of assets*	Aligned	Aligns with naming conventions, drawing standards, version-control workflows and exchange-format structures applicable during asset delivery.
ISO 704:2022 *Terminology work - Principles and methods*	Partially aligned	Central governance ensures consistent naming and reuse, but there is no formal semantic modelling or concept-system hierarchy as defined in ISO 704; future development may increase alignment.
ISO 16757-2:2018 *Data structures for electronic product catalogues for building services*	Partially aligned	ETIM MC is aligned with the standards principles for defining geometric volumes. It deviates by providing a single space “Functional space” that combines the purpose of the standards “Maintenance space” and “Operational space”.
ISO 16792:2021 *Technical product documentation - Digital product-definition data practices*	Partially aligned	ETIM MC follows ISO 16792 rules for 2D reference drawings (third-angle symbol, dimensioning, file naming & revision control) and provides the parametric data needed to generate a 3D model. It does not supply a neutral 3D dataset or PMI/GD&T, so full MBD requirements, model-authority metadata and archival formats remain out of scope.
ISO 23386:2020 *Governance of properties in data dictionaries*	Partially aligned	Governance exists at the class and feature level (via RFC process), but properties omit versioning, approval states and machine-readable metadata required by ISO 23386 for individual property governance for practical reasons.
ISO 23387:2020 *Data templates for construction objects*	Partially aligned	ETIM MC structures resemble data templates, but there is no semantic harmonization, constraint logic or machine-readability enforcement as defined in ISO 23387.

9. Document changelog

Version	Remarks
2.0-2025 (en)	Complete overhaul; New illustrations, restructure of content and chapters. Inclusion of new decisions, removal of outdated information, implementing appendixes into main document and clarify the scope. Compliance review for external international standards and transparency.
1.1-2023 (en)	Remove version nr from the drawing template.
1.0-2022 (en)	Complete overhaul, new illustrations, texts rewritten. New corporate design format Added new chapter on connection type classes Highlighted remarks removed. Added remark 1.3 about extending the ETIM guidelines Updated paragraph 2.2 Updated first paragraphs in chapter 4 Moved “4.1 versioning management” to new chapter nr. 5 and updated. Removed “Advice to PIM software developers” in 4.5.1 as this is no longer valid because of new versioning system.
0.6-2020 (en)	First English version. Major update on almost every page, based on first read-through of Marc Habets, and remarks of Heiko Dehne. Changes marked in red and yellow. Discussion topics highlighted with remarks.
1.0- 2016 (nl)	First Dutch version

Appendix 1: Guideline glossary

Term	Definition
2D reference drawing	A standardized SVG-based illustration that defines spatial dimensions and drawing codes for a Modelling Class.
° (Degrees)	A unit of angular measurement used in angle dimensioning (e.g., a1).
bSDD (buildingSMART Data Dictionary)	An IFC-compatible platform where ETIM concepts are mapped for BIM integration.
C (Coordinate)	A feature type used to define spatial positions in X, Y, Z dimensions (Drawing Code prefix: CA).
CA (Coordinate Assignment)	A feature indicating the exact spatial position of a component or connection point in X, Y, Z space.
CMT (Classification Management Tool)	Platform for managing features, values for both ETIM standards and the ETIM Product Classification classes.
Connection Type Class (CT)	A reusable class defining standardized connection interfaces (e.g., thread, flange), used across multiple Modelling Classes.
Drawing code	A coded reference (e.g., X1, d2, a3) assigned to specific dimensions or features in a reference drawing to support parametric modelling.
Drawing template	The standardized layout and format applied to all 2D reference drawings, including title block, projection symbol, and axis markings.
Dynamic Release	A release mechanism allowing newly approved RFCs to be published between major release cycles for early implementation.
DV (Directional Vector)	A vector feature that indicates the orientation of a connection or feature along the X, Y, Z axes.
EC (ETIM Product Class)	The product-level classification used for technical and commercial specification.
ETIM EC	ETIM Product Classification standard.
ETIM MC	ETIM Modelling Classification standard.
Feature code	A unique identifier for a property within a Modelling, Connection Type or Product Class.
Feature type	The classification of a parametric feature in terms of data format (e.g., Numeric, Logical, Coordinate, Matrix, etc.).
Inch	The standard imperial unit of length used in all dimensional features and coordinates.
LOD 200	A BIM Level of Detail standard signifying schematic, parametric geometry with general dimensions but without full fabrication details.
MC (Modelling Class)	A class defining standardized geometric and parametric features for a product category.
MG (Modelling Group)	A classification group in the MMT used to logically organize Modelling Classes independent of ETIM Product Class groupings.

mm (Millimetre)	The standard metric unit of length used in all dimensional features and coordinates.
MMT (Modelling Management Tool)	The platform used to create, manage, and govern Modelling Classes and Connection Type Classes.
Nominal size	A standardized size label (e.g., DN15) used for compatibility, but not defining the exact physical dimensions in ETIM modelling.
Optional feature	A geometric or functional element that may or may not be included in a product model, marked in drawings with parentheses.
Origin	The reference point (0,0,0) from which all coordinate and directional measurements are based in a Modelling Class.
PortCode	A numeric identifier that designates and distinguishes each connection point on a model.
RFC (Request for Change)	A formal submission process for proposing new classes or modifications to existing ones, including changes to drawings, dimensions, or feature sets.
SVG (Scalable Vector Graphics)	A vector file format used for storing ETIM reference drawings.

Appendix 2: Drawing compliance checklist

Use this checklist to validate each parameter reference drawing before final approval or submission. All requirements must be fulfilled unless marked as not applicable with justification.

	Yes	No
A. General drawing structure		
Drawing title matches the Modelling Class name exactly	☐	☐
Drawing file name follows the standard format (MCxxxxxx_Vx.svg)	☐	☐
Drawing is saved in SVG format and contains only vector graphics	☐	☐
Third-angle projection symbol is included	☐	☐
B. Views and layout		
Drawing includes at least two orthographic views (three recommended)	☐	☐
View orientation follows standard arrangement (Front, Top, Side)	☐	☐
Non-standard views (e.g., sections, details) are clearly labelled	☐	☐
Hidden lines are used appropriately for non-visible features	☐	☐
Coordinate axis and origin are shown using correct styles and colours	☐	☐
C. Dimensioning and drawing codes		
Drawing codes are assigned and placed consistently	☐	☐
Dimension lines include arrowheads, elbowed leaders, and correct layout	☐	☐
No redundant or conflicting dimensions are included	☐	☐
Nominal sizes (e.g., DN) are shown as labels only, not used for geometry	☐	☐
D. Linework, symbols, and geometry		
Correct line styles are used (solid, dashed, chain, phantom, etc.)	☐	☐
Hatching is applied when the object is recessed, buried, or mounted	☐	☐
Connector symbols and PortCodes are shown accurately	☐	☐
Optional features marked using parentheses and explained in the note	☐	☐
E. Text and labelling		
Fonts are correct and text sizes follow format rules	☐	☐
All text is in English and free from abbreviations (unless standardized)	☐	☐
Annotations are clear, horizontally aligned, and do not overlap	☐	☐
F. Content and parametric validity		
Minimum set of critical dimensions is defined to describe the geometry	☐	☐
Dimensions represent full physical size only where needed	☐	☐
Drawing reflects the expected level of detail	☐	☐
Drawing contents match the Modelling Class scope and naming convention	☐	☐