INTERNATIONAL STANDARD

ISO 16100-5

First edition
2009-03-01

Industrial automation systems and integration — Manufacturing software capability profiling for interoperability —

Part 5:
Methodology for profile matching using multiple capability class structures

Systèmes d'automatisation industrielle et intégration — Profil d'aptitude du logiciel de fabrication pour interopérabilité —

Partie 5: Méthodologie pour la correspondance de profil utilisant des structures de classe de capacité multiple

Reference number
ISO 16100-5:2009(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 16100-5 was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 5, *Architecture, communications and integration frameworks*.

ISO 16100 consists of the following parts, under the general title *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability*:

— *Part 1: Framework*

— *Part 2: Profiling methodology*

— *Part 3: Interface services, protocols and capability templates*

— *Part 4: Conformance test methods, criteria and reports*

— *Part 5: Methodology for profile matching using multiple capability class structures*

The following part is planned:

— *Part 6: Interface services and protocols for matching profiles using multiple capability class structures*

Introduction

The motivation for this part of ISO 16100 stems from the industrial and economic environment noted by TC 184/SC 5 in its strategic planning discussions, in particular:

— a growing base of vendor-specific solutions;

— user difficulties in applying standards;

— the need to move to modular sets of system integration tools;

— the recognition that application software and the expertise to apply that software are assets of the enterprise.

ISO 16100 is an International Standard for the computer-interpretable and human-readable representation of a software capability profile. Its goal is to provide a method to represent the capability of manufacturing software relative to its role throughout the life cycle of a manufacturing application, independent of a particular system architecture or implementation platform. This can lead to reduced production and information management costs to users and vendors/suppliers of manufacturing applications.

Certain diagrams in this part of ISO 16100 are constructed following unified modeling language (UML) conventions. Because not all concepts embodied in these diagrams are explained in the text, some familiarity with UML on the part of the reader is assumed.

Annex A describes the process for generating a manufacturing domain model (MDM) and manufacturing domain data (MDD).

Annex B gives an example of profile matching using multiple capability classes.

Industrial automation systems and integration — Manufacturing software capability profiling for interoperability —

Part 5: Methodology for profile matching using multiple capability class structures

1 Scope

This part of ISO 16100 specifies the methods and rules for matching existing manufacturing software unit (MSU) capability profiles and required capability profiles derived from multiple capability class structures. The methods and rules allow MSUs in manufacturing applications to be evaluated for interoperability and, in some cases, for interchangeability.

The following are outside the scope of this part of ISO 16100:

— services for creating, registering and accessing the various templates for the reference manufacturing domain models, the manufacturing domain data and the capability class structures;

— the conformance tables that reference Type 1 services defined and specified in ISO 16100-3;

— additional services needed for managing these templates in a database or equivalent object repository.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 16100-1, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 1: Framework*

ISO 16100-2:2003, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 2: Profiling methodology*

ISO 16100-3:2005, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 3: Interface services, protocols and capability templates*

ISO 16100-4:2006, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 4: Conformance test methods, criteria and reports*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 16100-1, ISO 16100-2, ISO 16100-3, ISO 16100-4 and the following apply.

3.1
capability class
element within the capability profiling method that represents software unit functionality and behaviour with regard to the software unit's role in a manufacturing activity, as denoted in a capability inheritance structure and as deployed in a capability aggregation structure

NOTE 1 The role of an MSU changes when used in different manufacturing activities; however, the MSU's corresponding capability class is positioned uniquely in an inheritance structure, but can assume different positions in an aggregation structure.

NOTE 2 In this part of ISO 16100, a capability class template is identical to a capability template (see ISO 16100-2:2003, 6.3, for requirements for capability templates).

NOTE 3 Adapted from ISO 16100-2:2003, 3.3.

3.2
capability class structure template
extensible markup language (XML) schema representing a hierarchy of capability classes

3.3
manufacturing domain data
unified modeling language (UML) class representing information about manufacturing resources, manufacturing activities, or items exchanged among manufacturing resources within a particular manufacturing domain

3.4
manufacturing domain data template
eXtensible markup language (XML) schema representing a manufacturing domain data

3.5
manufacturing domain model
particular view of a manufacturing domain, consisting of manufacturing domain data and relationships among them, corresponding to the domain's applications

3.6
manufacturing domain model template
eXtensible markup language (XML) schema representing a manufacturing domain model

4 Abbreviated terms

CCS	Capability Class Structure
CSI	Conformance Statement for the Implementation
MDD	Manufacturing Domain Data
MDM	Manufacturing Domain Model
MES	Manufacturing Operations Management
MSU	Manufacturing Software Unit
UML	Unified Modeling Language
XML	eXtensible Markup Language

5 Multiple CCSs referenced in manufacturing applications and in MSUs

5.1 Profile matching concept

Figure 1 shows the concept of the profile matching using multiple capability class structures.

NOTE 1 The capability class structures of the provided MSUs (left side of figure) are assumed to be based on the existence of a common capability class inheritance tree.

NOTE 2 The actual process for matching profiles uses the same algorithms that exist for matching XML schemas.

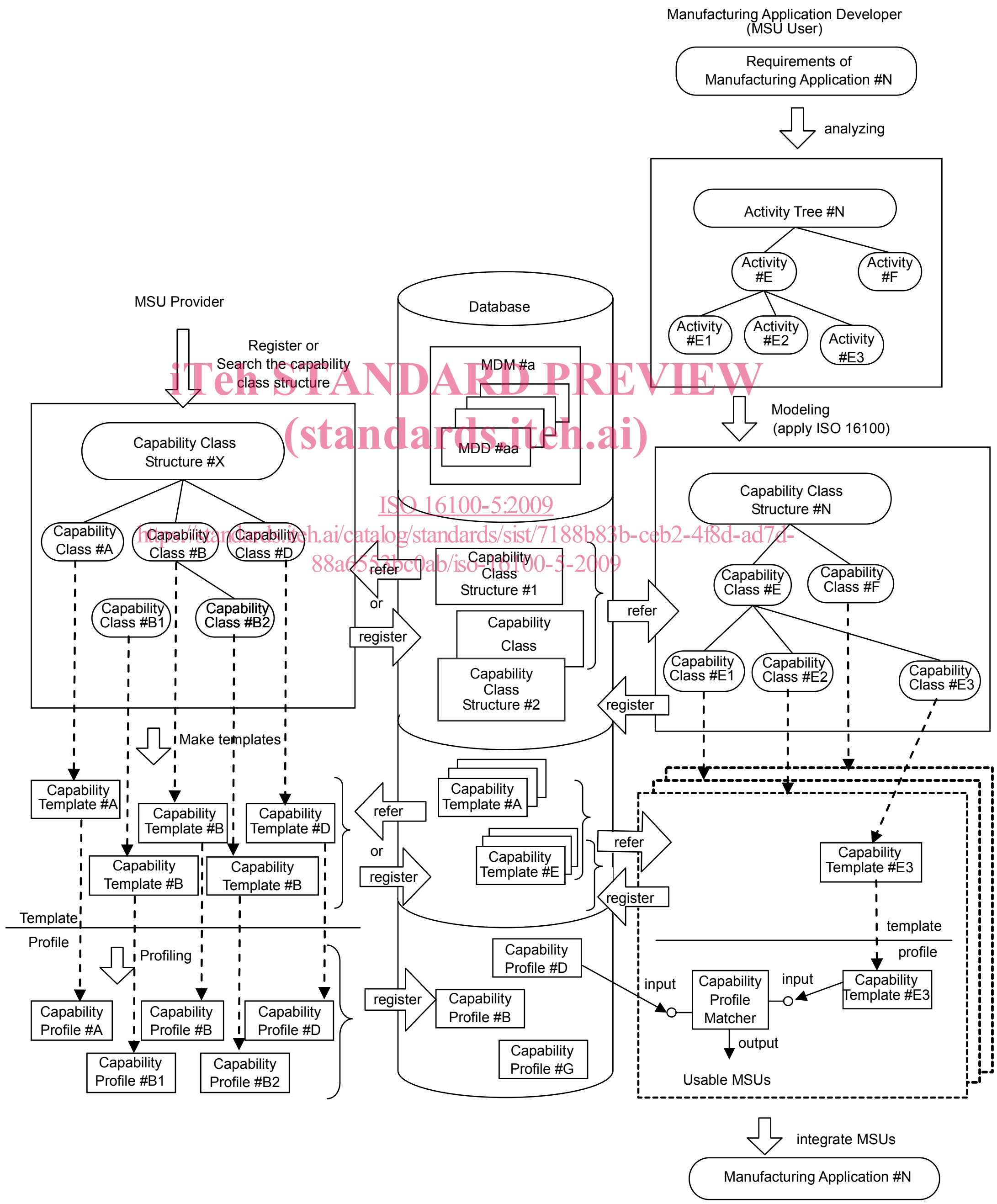

Figure 1 — Concept of profile matching using multiple capability class structures

5.2 Reuse of MSUs

To increase the efficiency of manufacturing application development, MSUs previously deployed in a similar manufacturing application should be reused. The MSU user shall only consider as candidates for reuse those MSUs whose capability profiles (see left side of Figure 1) meet his defined matching criteria for its capability profile (see right side of Figure 1). This part of ISO 16100 specifies a profile matching process wherein the capability class structures of the profiles being matched can be different, thus resulting in a larger number of candidate MSUs for reuse.

5.3 Registration of MSUs

A MSU provider registers an MSU so that it is widely available to many potential users of the MSU. The MSU provider shall perform the following procedures to prepare and register an MSU.

a) Analyze the set of activities that the MSU enables. The MSU can enable one or more activities.

b) Identify the capability class corresponding to each activity and search for the associated CCS to which the capability class belongs. If an MSU provides capabilities for two or more activities, those activities can belong to the same CCS or the different CCS.

c) Select the capability template for each capability class identified.

d) If there is no suitable CCS, construct the appropriate CCS and register it using appropriate database management methods. Then, generate the corresponding template and register that similarly.

e) Create the MSU capability profile by filling in the template(s) selected in procedure (c) or the new template(s) created in procedure (d), and register the template(s) using appropriate database management methods.

5.4 Applying reusable MSUs to meet requirements of a new manufacturing application

When a new manufacturing application is developed, the MSU user shall perform the following procedures.

a) Analyze the functional capability requirements of the manufacturing application and create an activity tree.

b) Create a CCS using existing or new capability classes to match the activity tree created in procedure (a), or select an existing CCS by using the methodology of ISO 16100-2.

c) For each capability class in the created or selected CCS, fill in the corresponding capability class template to create the set of required capability profiles.

d) Using a Type 2 capability profile matcher as described in ISO 16100-3, compare the set of required capability profiles to the available set of MSU capability profiles to find a set of existing MSUs that matches the set of required capability profiles.

e) Select the set of existing MSUs that meets the requirements of the new manufacturing application.

f) If the set of MSUs that meets the requirements is not found, develop a set of missing MSUs.

g) Combine the set of reused MSUs from procedure (e) and any set of developed MSUs from procedure (f) to meet the requirements of the new manufacturing application.

Figure 2 shows the implementation of the concept in Figure 1 to develop a new manufacturing application.

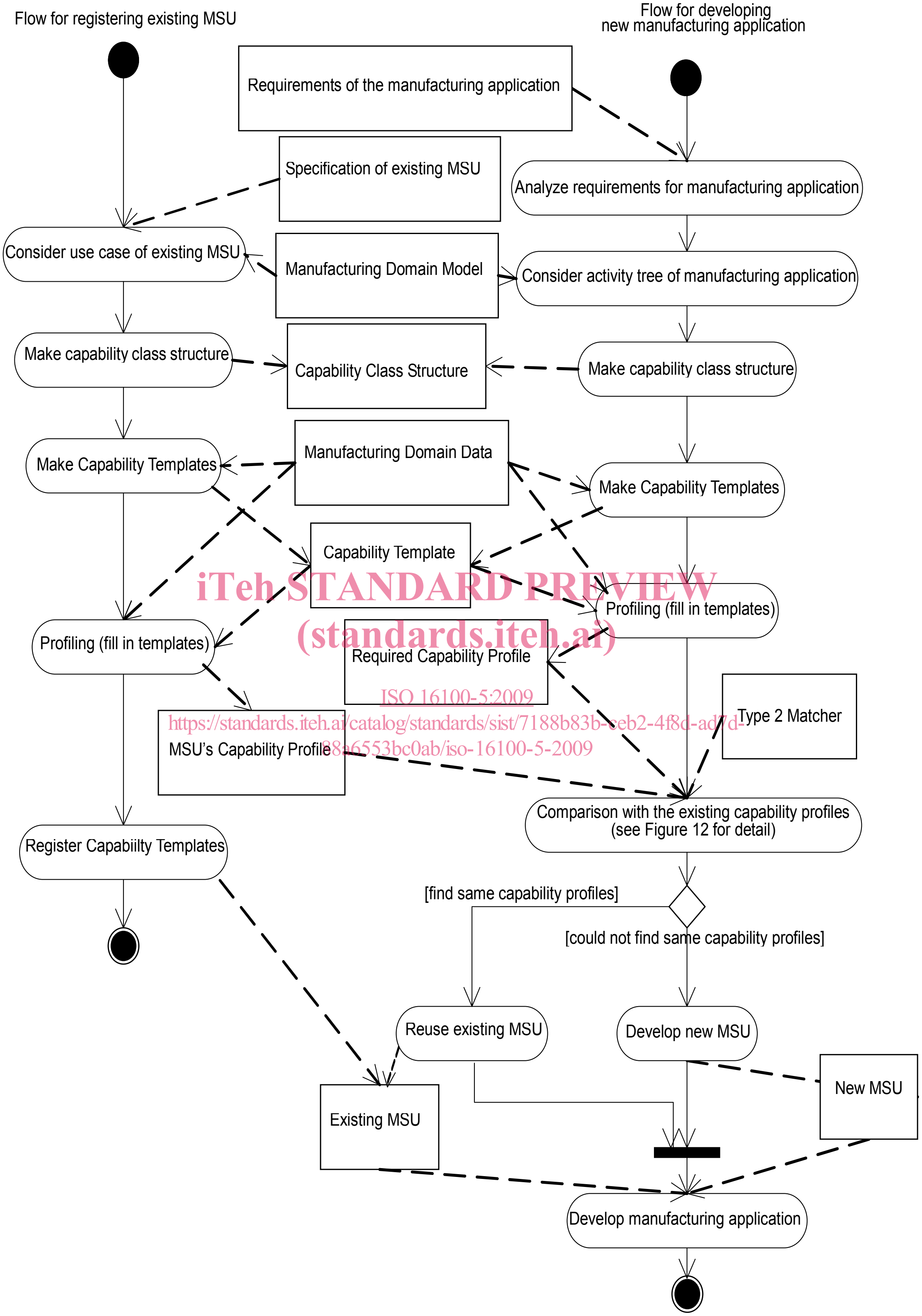

Figure 2 — Application development process with capability template, capability profile, and CCS

5.5 Manufacturing Domain Data

Figure 3 shows the MDD, MDM, CCS, Capability Class and their associations with the other manufacturing domain elements.

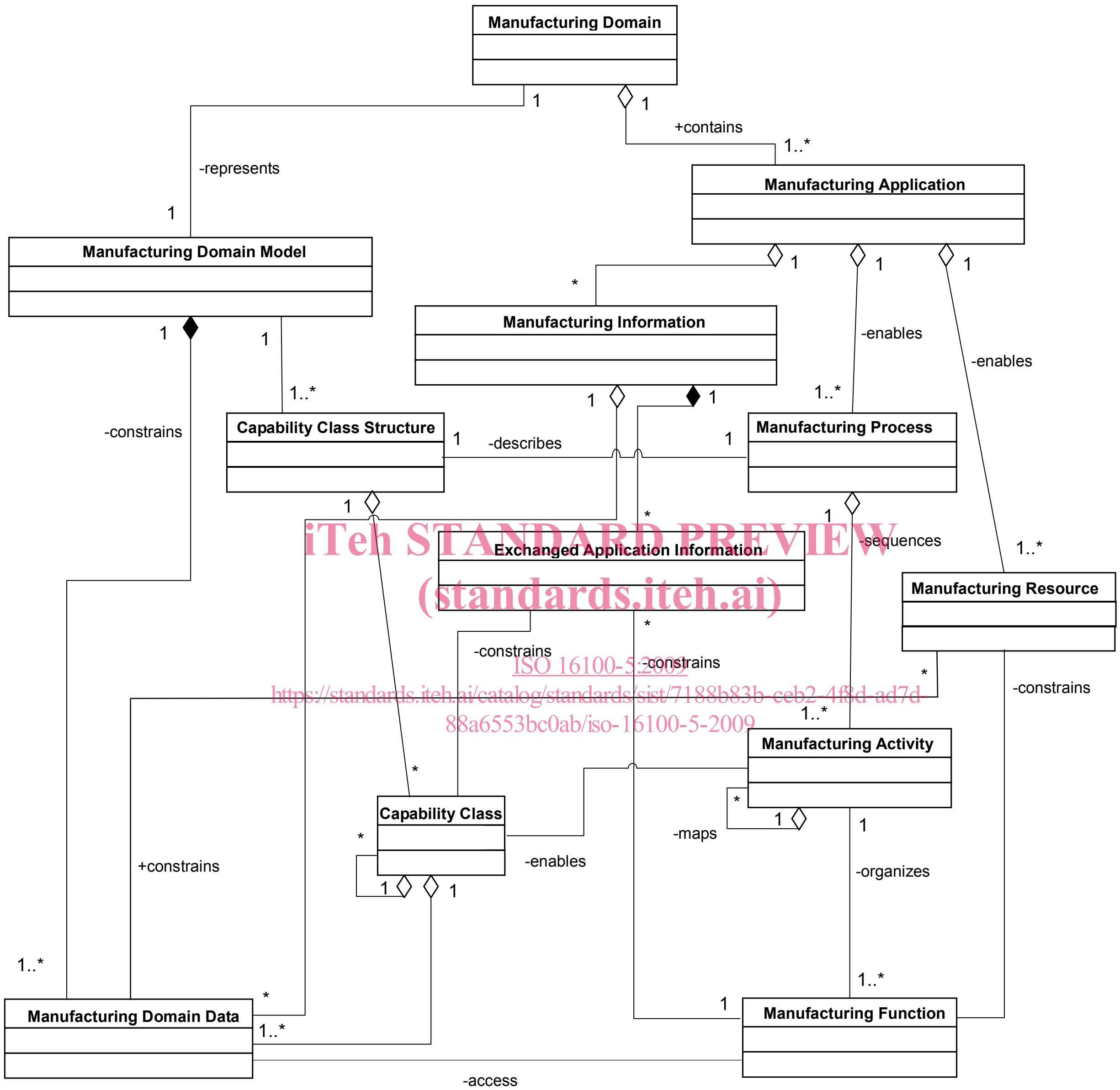

Figure 3 — Relationship between manufacturing domain and manufacturing application

MDDs represent different types of manufacturing information, including those that are exchanged between the resources within an application and between applications.

Figure 4 shows an example of a structure of a MDM with multiple MDDs. The process an MDM creator follows to create an MDM and MDDs is described in Annex A.

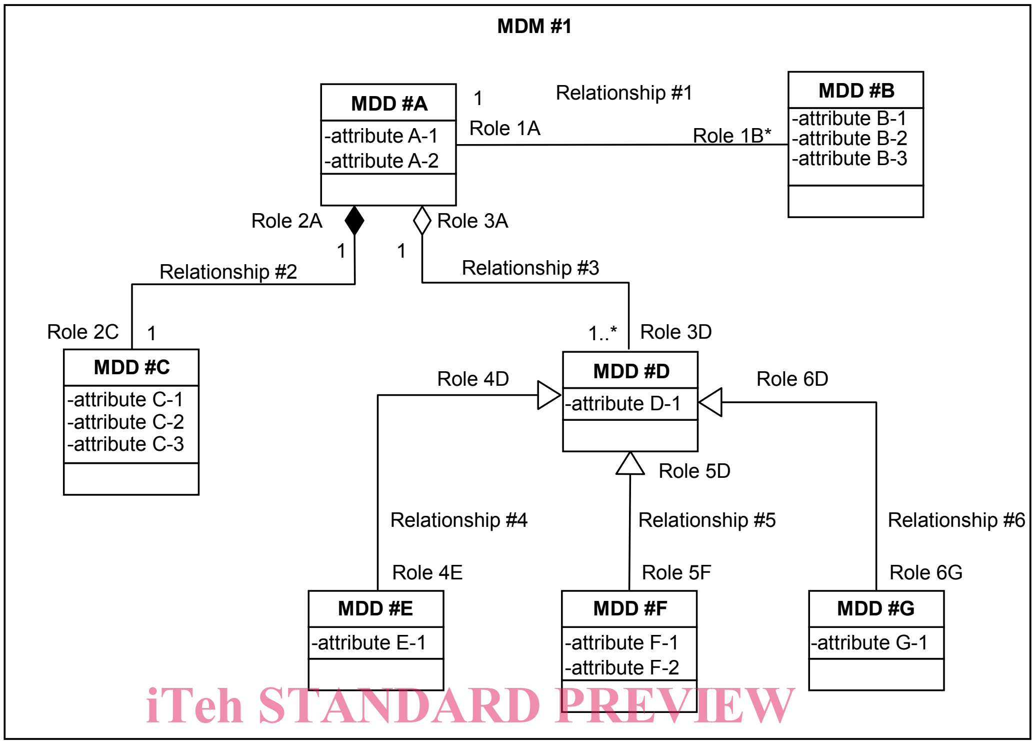

Figure 4 — Example structure of a MDM

Within a specific manufacturing domain, the MDM creator can represent a manufacturing application as a set of MDDs. An MDD provides information about various aspects of a manufacturing application such as:

a) manufacturing resources (e.g. MSU, equipment, automation devices, personnel, material, work-in-process inventory);

b) manufacturing processes (e.g. operations, activities);

c) manufacturing information exchanged (e.g. product data, recipe, manufacturing data, quality data);

d) relationships among the resources, processes and information exchanged.(e.g. data flow, network configuration, work flow).

In Figure 5, each MDD within a specific manufacturing domain consists of attributes and a set of relationships with other MDDs in the same domain using a relationship class. The relationship constraint and relationship type attributes in the relationship class delineate the allowed relationships among the MDDs within the specific MDM. The MDM creator shall descriptively name the MDD exchanged among manufacturing functions or among manufacturing activities such that each MDD is unique in the target manufacturing domain.

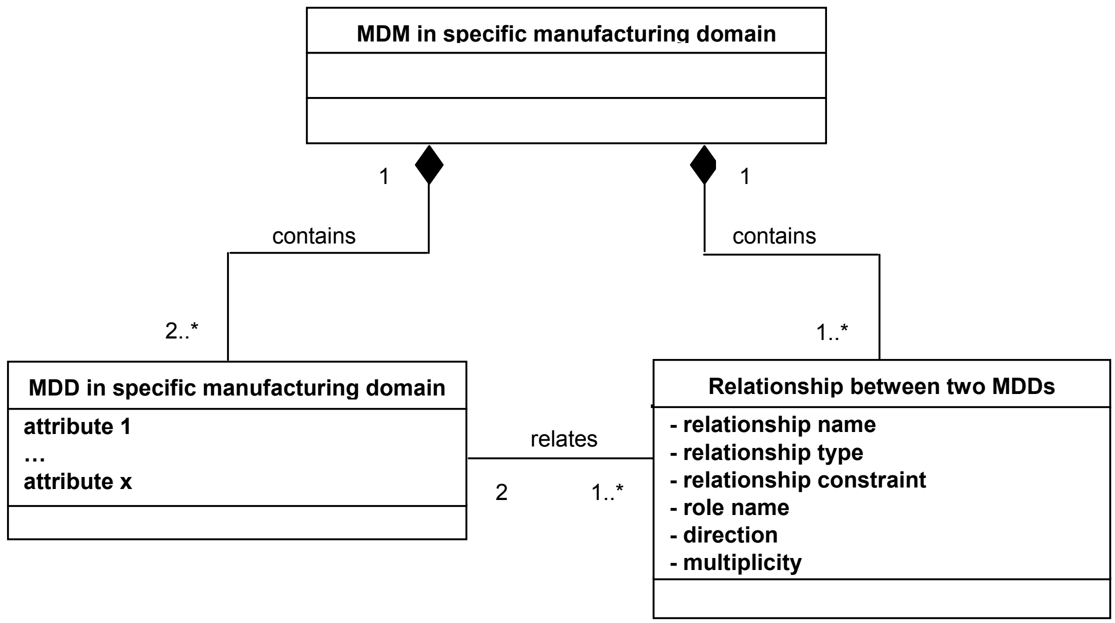

Figure 5 — Relationship between MDD and MDM

Figure 5 shows the relationship between a MDM and a minimum of two MDDs. These MDDs can typically form tree structures and have their relationships constrained by the relationship class noted in Figure 5. There is also the possibility of implementing specific class definitions specified as external classes from some related standards applicable to a particular manufacturing domain.

5.6 Mapping capability classes to MDDs

The MSU provider or MSU user models his activity tree based on the MDM from the requirements of the manufacturing application. The MSU provider or MSU user shall, in order to distinguish a particular activity in an activity tree, label an activity with an unambiguous and unique name, along with semantic information expressed in terms of a sequence of MDDs. The activities in the activity tree form the CCS. MSU providers and manufacturing application developers specify capability classes using a common set of MDDs.

Figure 6 shows two different CCSs mapped from their respective manufacturing activity trees. CCS #1 and CCS #2 are distinct structures, with some of the capability classes in the two CCSs being the same. These same capability classes can be recognized when a capability profile corresponding to a capability class is described using MDDs from the same MDM.

Each capability class in a structure is formed based on the services of a MDD or combined MDDs in the MDM. The canonical expression of a capability class includes specific lists of attributes, methods, and resources.

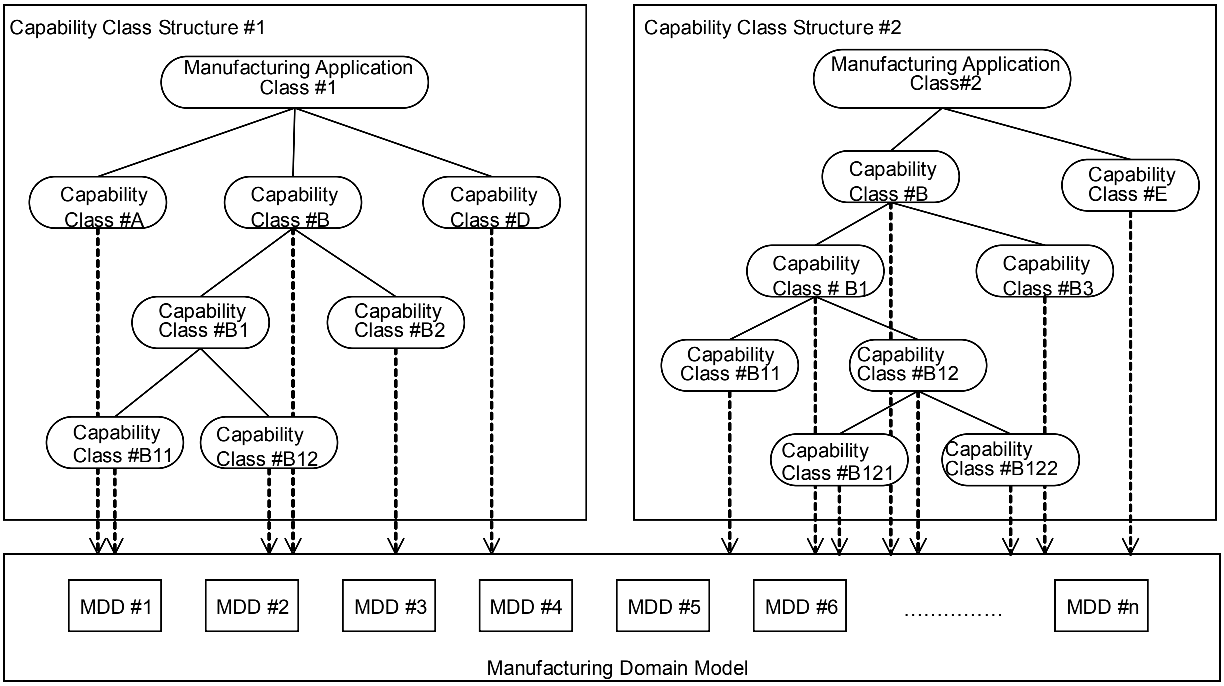

Figure 6 — Multiple capability classes described using MDDs from the same MDM

6 Methods and rules for capability profiling

6.1 Templates for MSU capability profiling

The following four templates are used in MSU capability profiling when there are multiple capability class structures:

a) CCS template;

b) Capability template;

c) MDM template;

d) MDD template.

6.2 CCS template

6.2.1 Conceptual structure

The CCS template shall contain, at a minimum, the following elements:

a) CCS Creator Name;

b) CCS ID;

c) ID for each capability class;

d) Parent Node ID for each capability class (the root node has no value for its Parent Node ID);

