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

Part 6:

**Interface services and protocols for
matching profiles based on multiple
capability class structures**

ISO 16100-6:2018
<https://standards.iteh.ai/catalog/standards/sist/d13c06cc-abe2-4aff-8dfb-7181e1ed18c4/iso-16100-6-2018>

*Systèmes d'automatisation industrielle et intégration — Profil
d'aptitude du logiciel de fabrication pour interopérabilité —
Partie 6: Services d'interface et protocoles pour la correspondance des
profils fondés sur les structures de classe d'aptitude multiple*



iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO 16100-6:2018

<https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018>



COPYRIGHT PROTECTED DOCUMENT

© ISO 2018

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	1
3 Terms and definitions	2
4 Symbols and abbreviated terms	3
5 Service provider interface services	4
5.1 Service sets	4
5.2 ESI service set	5
5.3 Dictionary Import Service Interface	6
5.3.1 Parts libraries imported to a repository	6
5.3.2 Relationship between parts libraries and MDDs	6
5.3.3 Mapping from a PLIB element to an MDD	7
6 Extended Service Interface	7
6.1 CPTI Group services	7
6.1.1 Scenarios handled by the CPTI Group	7
6.1.2 Capability profile template creation	7
6.1.3 The <i>accessTemplate</i> service	10
6.1.4 The <i>modifyTemplate</i> service	11
6.1.5 The <i>validateTemplate</i> service	12
6.1.6 The <i>deleteTemplate</i> service	12
6.2 The Extended CPI Group	13
6.2.1 Scenarios handled by the Extended CPI Group	13
6.2.2 Capability profile creation	14
6.2.3 The <i>accessProfile</i> service	15
6.2.4 The <i>modifyProfile</i> service	16
6.2.5 The <i>validateProfile</i> service	17
6.2.6 The <i>deleteProfile</i> service	18
6.3 The CCSI Group	19
6.3.1 Scenarios handled by the CCSI Group	19
6.3.2 Capability class structure creation	19
6.3.3 The <i>accessCCS</i> service	21
6.3.4 The <i>modifyCCS</i> service	22
6.3.5 The <i>validateCCS</i> service	22
6.3.6 The <i>deleteCCS</i> service	23
6.4 The Extended Matcher Group	24
6.4.1 Scenarios handled by the Extended Matcher Group	24
6.4.2 The <i>ExtendedMatcher</i> service	24
6.4.3 Matcher conformance testing	25
7 Formal ESI protocol description	28
7.1 The Extended Matcher Group	28
7.2 CPTI Group service protocol	29
7.2.1 Capability profile template creation	29
7.2.2 Capability profile template access	30
7.2.3 Capability profile template modification	30
7.2.4 Capability profile template conformance test	31
7.2.5 Capability profile template deletion	32
7.3 Extended CPI Group service protocols	32
7.3.1 Capability profile creation	32
7.3.2 Capability profile access	33
7.3.3 Capability profile modification	34
7.3.4 Capability profile conformance test	35

7.3.5	Capability profile deletion.....	35
7.4	CCSI Group service protocols.....	36
7.4.1	Capability class structure creation.....	36
7.4.2	Capability class structure access.....	37
7.4.3	Capability class structure modification.....	37
7.4.4	Capability class structure conformance test.....	38
7.4.5	Capability class structure deletion.....	38
7.5	Extended Matcher Group service protocols.....	39
8	Dictionary import service and protocol.....	40
8.1	The <i>DictionaryImporting</i> service.....	40
8.2	The <i>DictionaryImporting</i> protocol.....	40
Annex A	(informative) The capability model with MDDs.....	42
Annex B	(informative) Simplified matching of capability profile templates.....	47
Annex C	(informative) The profiles based on the capability profile templates.....	56
Annex D	(informative) The procedure for generating a capability class structure.....	59
Annex E	(informative) Mapping Parts Library (PLIB) to MDDs.....	60
Annex F	(informative) Mapping OTD to MDDs.....	63
Annex G	(informative) The procedure for matching two profiles.....	66
Bibliography	71

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO 16100-6:2018

<https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018>

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 5, *Interoperability, integration, and architectures for enterprise systems and automation applications*.

This second edition cancels and replaces the first edition (ISO 16100-6:2011), of which it constitutes a minor revision. The changes compared to the previous edition are as follows:

- in 5.2 b) 5) and b) 6), changed “a requirement capability profile or a requirement capability profile” to “an MSU capability profile or a requirement capability profile”;
- in 6.2.2.2.2, changed “*createTemplate* service” in the second sentence to “*createProfile* service”;
- in Figure 8, NOTE 2, changed “unique template identifiers” to “unique profile identifiers”;
- in 6.2.5, changed “*returnTestingResult* services” to “*returnTestResult* services”;
- in 7.3.5, changed “The *deleteTemplate* service deletes an existing template” to “The *deleteProfile* service deletes an existing profile”;
- in 7.4.1.2, changed “generates a template” to “generates a CCS”;
- in 7.4.5, changed “deletes an existing template” to “deletes an existing CCS”;
- in Annex A, changed “the class capability model” to “the capability class model”;
- in Annex G, added double quotation marks for object names;

A list of all parts in the ISO 16100 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

The motivation for ISO 16100 stems from the industrial and economic environment, in particular:

- a growing base of vendor-specific solutions;
- user difficulties in applying standards;
- need to move to modular sets of system integration tools;
- 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 capability profile. Its goal is to provide a method to represent the capability of manufacturing application 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 document are constructed following 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.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO 16100-6:2018](https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018)

<https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018>

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

Part 6: Interface services and protocols for matching profiles based on multiple capability class structures

WARNING — This document provides a specification intended to be implemented in software. Incompatibilities may result in machine-to-machine communication in the case of software developed on the basis of translations of this document into languages other than the official ISO languages. Accordingly, any implementations should be developed only on the basis of the texts in the official ISO languages.

1 Scope

This document defines the detailed interface services and protocols used in a matching method based on multiple capability class structures. This document also defines a CPTI (Capability Profile Template Interface) Service Group, an Extended CPI (Capability Profile Interface) Service Group and an Extended Matcher Interface Service Group, which is extensions of the Type 1, Type 2 and Type 3 services, respectively, specified in ISO 16100-3:2005, 5.4.

This document also defines the CCSI (Capability Class Structure Interface) Service Group, an additional service group used to create, register, access and modify a capability class structure for the reference manufacturing domain models specified in ISO 16100-5:2009, Clause 6.

This document also specifies detailed contents of the specific part of a capability profile template defined in ISO 16100-5:2009, Clause 7.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 16100-1:2009, *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*

ISO 16100-5:2009, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 5: Methodology for profile matching using multiple capability class structures*

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, ISO 16100-5 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

3.1 capability class
<software unit 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

Note 1 to entry: 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 it can assume different positions in an aggregation structure.

Note 2 to entry: In this document, a capability class template is identical to a capability profile template (see ISO 16100-2:2003, 6.3 for requirements for capability templates).

Note 3 to entry: In general, a capability class maps to an activity. The capability class is distinct within a capability inheritance structure and can form a capability aggregation structure with other capability classes.

[SOURCE: ISO 16100-2:2003, 3.3, modified — The domain in angle brackets has been added; Note 1, 2 and 3 to entry have been added; the wording “software units role” has been changed to “software unit’s role”.]

3.2 capability class structure
hierarchy of capability classes

ISO 16100-6:2018
<https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018>

Note 1 to entry: This structure is intended for modeling capability aggregation hierarchies in the target domains of ISO 16100-1:2009, Figure 2.

3.3 capability class structure template
XML schema representing a capability class structure

[SOURCE: ISO 16100-5:2009, 3.2, modified — The full form for “XML” has been deleted; the wording “a hierarchy of capability classes” has been changed to “a capability class structure”.]

3.4 capability profile template
schema for a manufacturing software capability profile

3.5 extended service interface
set of service access points defined in this document that handle manufacturing domain data, manufacturing domain models, capability class structures, capability profiles and capability profile templates

Note 1 to entry: "Extended" refers to both the services specified in this document and the "basic" services specified in ISO 16100-3.

3.6**manufacturing domain data**

information, represented by a UML class, about manufacturing resources, manufacturing activities, or items exchanged among manufacturing resources within a particular manufacturing domain

[SOURCE: ISO 16100-5:2009, 3.3, modified — The wording “unified modeling language (UML) class representing information” has been changed to “information, represented by a UML class”.]

3.7**manufacturing domain data template**

XML schema representing a manufacturing domain data

[SOURCE: ISO 16100-5:2009, 3.4, modified — The full form for “XML” has been deleted.]

3.8**manufacturing domain model**

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

[SOURCE: ISO 16100-5:2009, 3.5]

3.9**parts library**

<manufacturing> collection of part descriptions or catalogue

Note 1 to entry: The term “parts library” also refers to a dictionary such as a PLIB dictionary in ISO 13584 or OTD in ISO 22745.

iteh STANDARD PREVIEW
(standards.iteh.ai)

4 Symbols and abbreviated terms

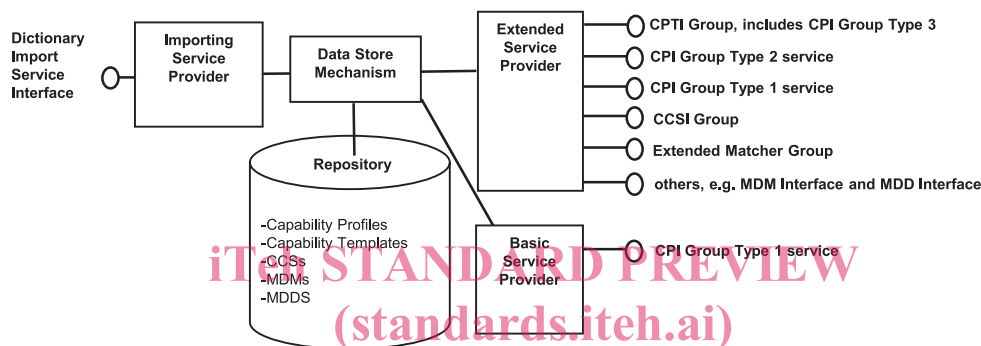
BSU	Basic Semantic Unit	https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018
CCS	Capability Class Structure	
CCSI	Capability Class Structure Interface	
CPI	Capability Profile Interface	
CPTI	Capability Profile Template Interface	
CSI	Conformance Statement for the Implementation	
ESI	Extended Service Interface	
ESP	Extended Service Provider	
ICD	International Code Designator	
MDD	Manufacturing Domain Data	
MDM	Manufacturing Domain Model	
MSU	Manufacturing Software Unit	
OTD	Open Technical Dictionary	
PLIB	Parts Library (as specified in ISO 13584)	
UML	Unified Modeling Language	

- URL Uniform Resource Locator
- URN Uniform Resource Name
- XML eXtensible Markup Language

5 Service provider interface services

5.1 Service sets

Figure 1 shows all the services and their relations to Extended Service Providers and Basic Service Providers to handle capability profiles, capability profile templates, CCSs, MDMs, MDDs and MDD objects. Basic Service Providers handle CPI group Type 1 services. Extended Service Providers handle CPTI, CPI and CCSI services. In addition, Extended Service Providers support extended matcher services and other service interfaces to handle MDMs and MDDs.



NOTE 1 This figure is not in accordance with UML conventions. The line between the Data Store Mechanism and the Repository represents the rules for adding, removing and changing contents of the Repository. The line between the Data Store Mechanism and the Extended Service Provider represents a mapping of the extended services to the Data Store services. The mapping is typically implementation-specific and therefore not part of the scope of this document.

NOTE 2 The boldfaced elements in this figure are specifically addressed in this document.

NOTE 3 The contents in the Repository are stored as XML files.

NOTE 4 The ESI access point is represented elsewhere in this document by the object *ServiceAccessPoint*.

NOTE 5 The Type 1 CPI service group, which is briefly described in ISO 16100-3:2005, 5.4, includes Type 1 matcher service.

NOTE 6 The Type 2 CPI service group, which is briefly described in ISO 16100-3:2005, 5.4, does not include Type 2 matcher services, which are part of the Extended Matcher Group.

NOTE 7 The Type 3 CPI service group is briefly described in ISO 16100-3:2005, 5.4.

Figure 1 — Extended Service Provider service sets

All services have the following characteristics:

- a) when a service is conducted, there is one service provider and one service user, and no other third party is involved;
- b) the service user initiates all service invocations, which are distinct from the lower communications layer service invocations;
- c) a service user invocation is always accompanied by a response from the service provider;

- d) any service user invocation and the corresponding response(s) are conducted in a bounded time frame, as determined by the service user or the service provider or both;
- e) a service invocation at a service access point is processed when a response to a prior service invocation is completed;
- f) an invocation is made for a single service; there is no invocation for a service group;
- g) a service to register additions and updates to the repository uses the data store mechanism in [Figure 1](#);
- h) the state of an object in the repository is one of the following:
 - 1) stored: an object is stored in the repository after a creation request;
 - 2) registered: an object is registered into the repository after it is conformance tested;
 - 3) deleted: an object is deleted from the repository after a deletion request.

5.2 ESI service set

The generic ESI services provided by an ESP can be organized in specific combinations into a CPI Group Type 1 (see ISO 16100-3) and the following four service groups, which are described in more detail in [Clause 6](#). Other service groups can exist, e.g. MDM, MDDs and MDD objects, but are not defined in this document.

- a) CPTI Group, that includes CPI Group Type 3, allows the following services:
 - 1) create a new capability profile template;
 - 2) access a capability profile template;
 - 3) modify a capability profile template;
 - 4) conformance test a capability profile template;
 - 5) register an MSU capability profile;
 - 6) delete an MSU capability profile.
- b) CPI Group Type 2 (see ISO 16100-3) allows the following services:
 - 1) create a new MSU capability profile or a new requirement capability profile;
 - 2) access an MSU capability profile or a requirement capability profile;
 - 3) modify an MSU capability profile or a requirement capability profile;
 - 4) conformance test a capability profile;
 - 5) register an MSU capability profile or a requirement capability profile;
 - 6) delete an MSU capability profile or a requirement capability profile.
- c) CCSI Group allows the following services:
 - 1) create a new capability class structure;
 - 2) access a capability class structure;
 - 3) modify a capability class structure;
 - 4) conformance test a capability class structure;

- 5) register a capability class structure;
- 6) delete a capability class structure;
- d) Extended Matcher Interface Group allows the following services:
 - 1) access an MSU capability profile or a requirement capability profile;
 - 2) match two capability profiles, each referencing a different capability class structure.

Annex A provides information on the capability model with MDDs.

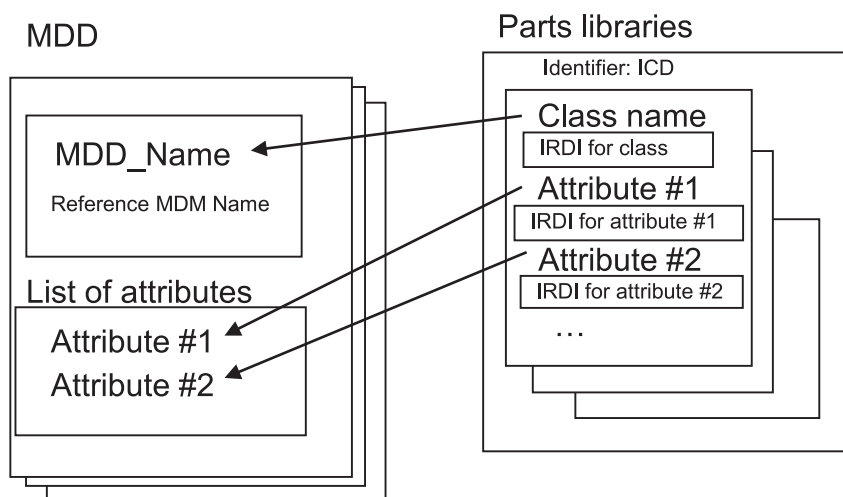
5.3 Dictionary Import Service Interface

5.3.1 Parts libraries imported to a repository

The Dictionary Import Service Interface enables the importing of a library (e.g. PLIB dictionary or OTD) to a repository.

5.3.2 Relationship between parts libraries and MDDs

A parts library imported to a repository can be used as part of MDDs in capability profiling. MDDs in an MDM include the definition of manufacturing activities. Since a PLIB dictionary and an OTD includes definitions of technical terms, it can be used in capability profiling in the same manner as MDDs are used. All classes and attributes in a PLIB dictionary or OTD can be identified by an unambiguous identifier in accordance with ISO/TS 29002-5 and ISO 22745-13. The unambiguous identifier is a form of international registration data identifier (IRDI) as specified in ISO/IEC 11179-5. Classes and attributes within a PLIB dictionary can also be specified by a combination of dictionary and a BSU, a code for each PLIB class and attribute internal to the dictionary. A BSU for an attribute in a PLIB dictionary is the combination of the attribute code and the BSU of the parent class. For an OTD, it is not necessary to reference the BSU of the parent class, since ISO 22745 is classification-neutral, and properties are defined independently of classes in an OTD. The relationship between a PLIB dictionary or OTD and an MDD is determined by mapping the unambiguous identifier for each MDD and attribute of an MDD, as shown in Figure 2.



NOTE 1 This figure is not in accordance with UML conventions.

NOTE 2 In ISO 13584 the term “property” is used instead of “attribute”.

Figure 2 — Relationship between parts libraries and MDDs

5.3.3 Mapping from a PLIB element to an MDD

The element "MDD_Name" in [Figure 2](#) shall have the following extended attributes: "dictionary id", "parent", "BSU", "version" and "revision". The attribute set of "dictionary id", "parent" and "BSU" can be used as an identifier of the MDD.

According to [Figure 2](#), attributes in an MDD refer to attributes of items in a PLIB dictionary. An MDD corresponds to an element in a PLIB dictionary. Attributes belonging to the same element would be associated with one MDD.

[Annex E](#) provides information on mapping PLIB to MDDs.

6 Extended Service Interface

6.1 CPTI Group services

6.1.1 Scenarios handled by the CPTI Group

The CPTI Group handles the following capability profile template scenarios:

- a) create a capability profile template for a specific class structure, either by partially filling the generic formal structure of the capability profile template or by modifying an existing capability profile template with a new capability profile template ID using the MDDs in the MDM, and then receive a capability profile template from a service provider;
- b) request a capability profile template from a repository based on a capability profile template ID and then receive a capability profile template from a service provider;
- c) modify a capability profile template according to either user requirements or the results from a conformance test and receive a modified capability profile template from a service provider;
- d) test a capability profile template against the capability profile template conformance criteria found in ISO 16100-5:2009, Clause 8 and receive either a positive, negative or matching level response from a service provider;
- e) register a tested capability profile template into a repository and receive a "registered" status from the service provider;
- f) delete a capability profile template based on a capability profile template ID and receive a "deleted" status from a service provider.

[Annex B](#) provides information on the simplified matching of capability profile templates.

6.1.2 Capability profile template creation

6.1.2.1 The creation process

The process for creating a capability profile template for a capability class of a particular capability class structure consists of configuring a generic formal structure of a capability profile template by either

- a) filling in specific values to certain attributes of certain elements in the generic formal structure of the capability profile template as defined in ISO 16100-5:2009, 6.3, or
- b) modifying previously filled-in values in an existing capability profile template.

6.1.2.2 Capability profile template configuration

A generic formal structure of the capability profile template may be either partially or completely filled according to the application requirement.

For any capability profile template, the following attributes and elements shall be filled in or modified:

- a) attributes "id" and "name" in element "Capability Profile Template";
- b) attribute "domain_Name" in element "Reference_MDM_Name";
- c) attribute "format_name" in element "MDD_Description_Format" with one of the following four values:
 - 1) "Set_Of_MDD_Objects";
 - 2) "List_Of_MDD_Objects";
 - 3) "Time_Ordered_MDD_Objects";
 - 4) "Event_Ordered_MDD_Objects";
- d) attributes "name" and "action" in element "MDD_Name".

Each value filled in or modified for the attributes "id" in 6.1.2.2 a), "domain_Name" in 6.1.2.2 b) and "name" in 6.1.2.2 d) shall be unique.

A capability profile template shall be used to create a capability profile associated with a capability class.

[Annex B](#) provides information on the simplified matching of capability profile templates.

6.1.2.3 The *createTemplate* service

6.1.2.3.1 Template based on a generic formal capability structure

The *createTemplate* service shall allow a template user to create a template based on a generic formal capability structure. When creating a template based on a formal capability structure, the *createTemplate* service uses at a minimum the *requestBlankTemplate*, *returnBlankTemplate*, *processFilledTemplate* and *returnProcessingResult* services. The *createTemplate* service shall consist of the following steps:

- a) the template user invokes the *requestBlankTemplate* service of the *ServiceAccessPoint* object, in which there are no parameters associated with the *requestBlankTemplate*;
- b) the service provider invokes the *returnBlankTemplate* service of the *ServiceAccessPoint* object, in which the parameters of the *returnBlankTemplate* service are blank template and creation error;
- c) the template user fills in the blank template using MDDs of the MDM and then invokes the *processFilledTemplate* service of the *ServiceAccessPoint* object, in which the parameter of the *processFilledTemplate* service is template ID;
- d) the service provider checks the uniqueness of the template ID and then invokes the *returnProcessingResult* service of the *ServiceAccessPoint* object, in which the parameters of the *returnProcessingResult* service are ID check error and storage error.

[Figure 3](#), above the dotted line, provides a UML sequence diagram of the mandatory steps for the creation of a template from a formal template structure.

6.1.2.3.2 Template created by modifying an existing capability profile template

The *createTemplate* service shall allow a template user to create a template modified from an existing template. When creating a template modified from an existing template, the *createTemplate* service uses at a minimum the *requestExistingTemplate*, *returnExistingTemplate*, *processModifiedTemplate* and *returnProcessingResult* services. The *createTemplate* service shall consist of the following steps:

- a) the template user invokes the *requestExistingTemplate* service of the *ServiceAccessPoint* object, in which the parameter of the *requestExistingTemplate* service is template ID;

- b) the service provider invokes the *returnExistingTemplate* service of the *ServiceAccessPoint* object, in which the parameters of the *returnExistingTemplate* service are existing template and processing error;
- c) the template user modifies the existing template and then invokes the *processModifiedTemplate* service of the *ServiceAccessPoint* object, in which the parameter of the *processModifiedTemplate* service is template ID;
- d) the service provider checks the uniqueness of the template ID and then invokes the *returnProcessingResult* service of the *ServiceAccessPoint* object, in which the parameters of the *returnProcessingResult* service are ID check error and storage error.

[Figure 3](#), below the dotted line, provides a UML sequence diagram of the mandatory steps for the creation of a template from an existing template.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO 16100-6:2018](#)

<https://standards.iteh.ai/catalog/standards/sist/dd2c06ce-e6e2-4aff-8dfa-718fc1edf8c4/iso-16100-6-2018>