Manufacturing Software Interoperability Services which ISO 16100 brings about

Michiko Matsuda

Kanagawa Institute of Technology,
1030 Shimo-ogino, Atsugi-shi, Kanagawa, Japan
(ISO/TC 184/SC 5/WG 4 convenor)
matsuda@ic.kanagawa-it.ac.jp

Abstract. ISO 16100 series provides a standardized methodology for the interoperability of manufacturing software using capability profiling. The method for describing software capability as Capability Profile and the way for exchanging software capability as information through Capability Profile are provided by ISO 16100. By using these two fruits from ISO 16100, two new interoperability services are proposed. One is a new manufacturing application developing method using the manufacturing software capability catalogue as cloud service. The other is a new manufacturing system configuration method by plug-and-play of manufacturing software. Finally, further enhancement of interoperability services are discussed.

Keywords: manufacturing software interoperability, capability profiling, capability template, MSU(Manufacturing Software Unit), software integration.

1 Introduction

The ISO 16100 series which is titled ‘Manufacturing Software Capability Profiling for Interoperability,’ consists of six parts. Part 1 is titled ‘Framework [1].’ Part 2 is ‘Profiling methodology [2].’ Part 3 is ‘Interface services, protocols and capability templates [3].’ Part 4 is ‘Conformance test methods, criteria, and report [4].’ Part 5 is ‘Methodology for profile matching using multiple capability classes [5].’ The last: Part 6 is ‘Interface services and protocols for matching profiles based on multiple capability class structures [6].’ All parts of ISO 16100 have been published by spring of 2010. Now, all parts are ready for use. The ISO 16100 series enables manufacturing software integration by providing the followings: 1) standard interface specifications that allow information exchange among software in industrial automation systems developed by different vendors, 2) software capability profiling using a standardized method to enable users to select software that meet their functional requirements, and 3) a conformance test method that ensures the integrity of the software integration. The ISO 16100 methodology is also applicable and usable for developing general software applications and for describing capabilities of application software.

In this paper, two new directions and usages of ISO 16100 are proposed. One is a new manufacturing software developing methodology corresponding to globalization and the trend of the open system architecture, by using ISO 16100 as tools to find and use skillfully software units which are provided by various vendors and to reuse existing software units. The other is a new manufacturing system configuration methodology by using ISO 16100 as information exchange tools which have abilities for describing software capabilities and interface services. Finally, enhancements and extensions of the ISO 16100 methodology are discussed to proceed in the proposed directions.

2 Manufacturing Software Capability Profiling in ISO 16100

As a premise of a proposal, the methodology for manufacturing software capability profiling and elements for capability profiling in ISO 16100 are introduced [7].

2.1 Capability Profiling of Manufacturing Software Units

Capabilities of a manufacturing application are represented as an activity tree structure that is both nested and hierarchical. The activity tree is structured based on its associated manufacturing domain. To distinguish a particular activity in an activity tree, an activity has an unambiguous and unique name, along with semantic information expressed. The Capability Class Structure (CCS) is formed from the activities in the activity tree. As shown in Fig. 1, a CCS corresponds to the activity tree and a Capability Class is unique when the activity can be pointed to in the activity tree. An activity tree and a CCS have a one to one mapping. This means the activity tree is modeled as a CCS. The capability of a Manufacturing Software Unit (MSU) is expressed in terms of Capability Classes. At each level, the MSU is modeled as a set of Capability Classes organized in a similar structure. At the bottom level, the MSU is modeled as one Capability Class. These classes denote the combination of the manufacturing functions, resource, and information handled by the MSU according to the requirements of the manufacturing process.

A MSU that enables or supports an activity with an associated Capability Class is concisely described in a Capability Template. A Capability Template is unique when a Capability Class can be pointed to. In other words, one Capability Class has one corresponding Capability Template as shown in Fig. 1. This means that a Capability Template is a concrete class for a Capability Profile. MSU's Capability Profiles are obtained by filling adequate Capability Templates.

If a MSU vendor wants to widely distribute his developed MSU, the vendor makes several Capability Profiles for one MSU corresponding to each CCS. A MSU vendor registers a Capability Profile of a MSU so that it is widely available to many potential users of the MSU. When a manufacturing system configurator or manufacturing application developer wants to search proper MSUs, they create the Capability Profile by filling the Capability Template with the required capabilities and seek a matching of the required Capability Profile and MSU's Capability Profiles.

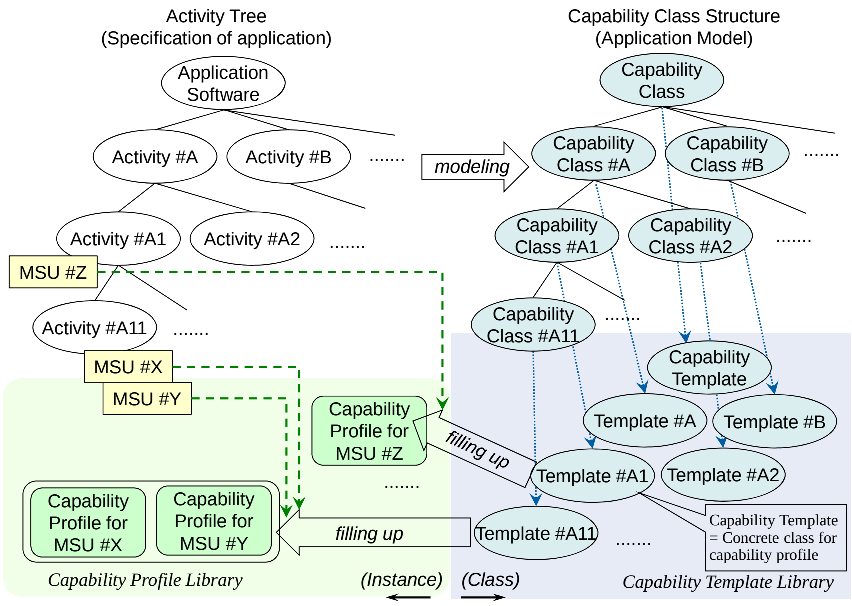

Fig. 1. Capability templates and capability profiles.

2.2 Elements for Capability Profiling

A manufacturing application in a specific domain is enabled by manufacturing processes, manufacturing information and manufacturing resources as shown in Fig. 2. A manufacturing process is composed on a set of manufacturing activities. A manufacturing process may have a nested or hierarchical structure of manufacturing activities. The sequence and schedule of functions performed is determined by the sequence and schedule of the activities that comprise a particular process. The MSUs deployed to perform the functions are considered to execute according to the required sequence and schedule of their associated functions. A MSU consists of one or more manufacturing software components, performing a definite function or role within a manufacturing activity while supporting a common information exchange mechanism with other MSUs. In this framework, MSU's capabilities are profiled using CCS and a Capability Template as mentioned in Sec. 2.1.

The relationships of important elements for capability profiling such as Capability Template and Capability Profile, and elements in the above mentioned framework are shown in Fig. 2. Furthermore, two new elements are introduced for capability profiling. One is the MDM: Manufacturing Domain Model. The other is the MDD: Manufacturing Domain Data. The MDM is a particular view of a manufacturing domain, consisting of MDDs and relationships among them, corresponding to the domain's applications. A set of MDDs works like a terminology set in the applicable domain.

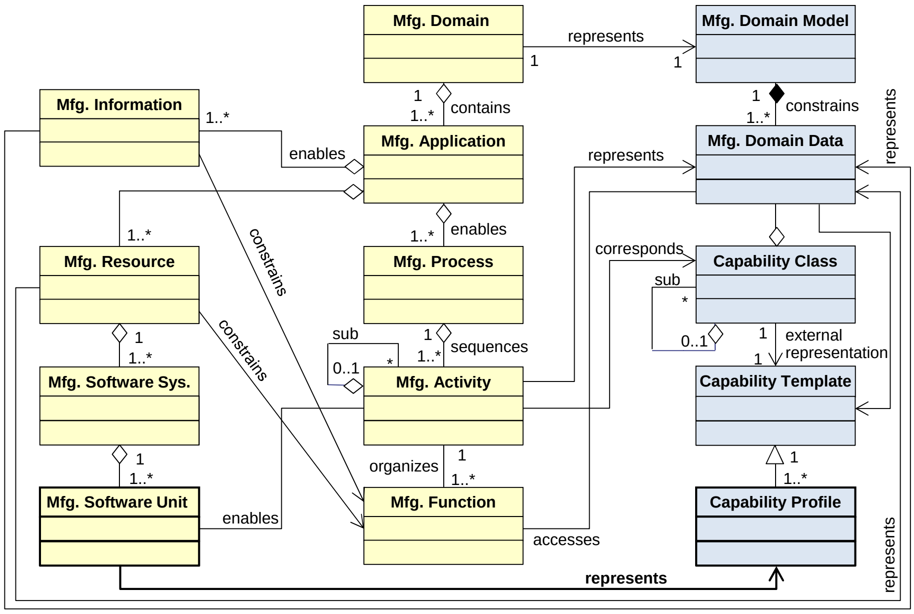

Fig. 2. Elements for capability profiling.

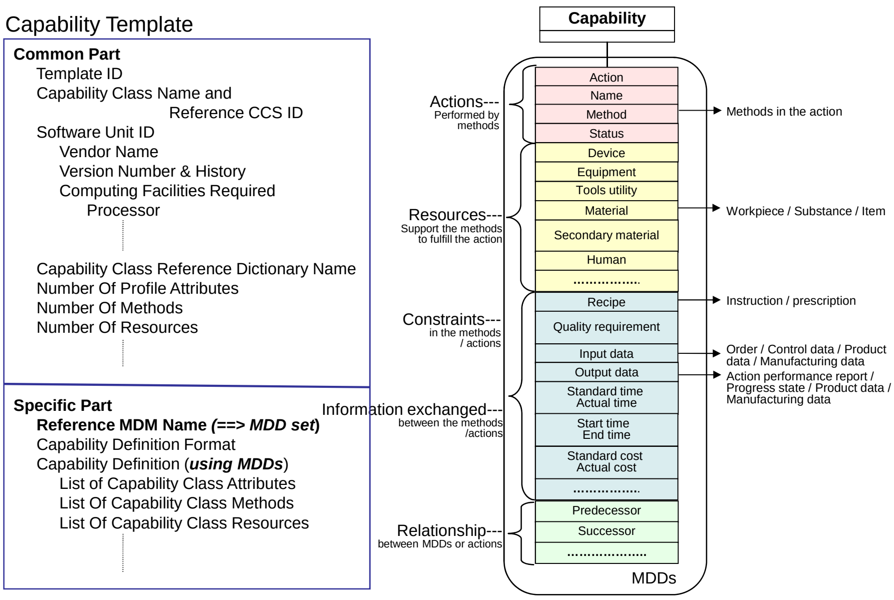

Fig. 3. Manufacturing domain data and a capability template.

An MDD represents information about various aspects of a manufacturing application such as manufacturing resources, manufacturing processes, manufacturing information exchanged, and relationships among the resources, processes and information exchanged as shown in the right part of Fig. 3 [8]. Each MDD within a specific manufacturing domain consists of attributes, operation types and a mapping between them. The MDD exchanged among manufacturing functions or among manufacturing activities is descriptively named such that each MDD is unique in the target manufacturing domain. A Capability Template shown in the left part of Fig. 3 contains a Common Part and a Specific Part. The Specific Part contains the elements: Reference MDM Name, list of MDD objects, and capability definition (e.g. time ordered access to MDD objects).

3 Interoperability Framework for Manufacturing Software

The Interoperability framework in ISO 16100 is summarized in Fig. 4 [7]. On the left the flow shows the procedure that a MSU vendor performs to prepare and register a Capability Profile of a MSU. System configurators or developers develop the manufacturing application by searching existing adequate MSUs and combining them. The flow on the right shows the procedure that system configurators or developers perform to develop a new manufacturing application.

The flow shown on the left of Fig. 4 shows the whole process for Capability Profile registration. When a Capability Profile is first registered, the adequate Capability Template selection is necessary. First, a suitable manufacturing domain for the registering MSU is chosen. Second, the applicable activity tree in the domain is adopted and the activity is identified in the tree. Third, the corresponding Capability Class is adopted. When a new activity tree is created, new Capability Classes are generated. Fourth, an adequate Capability Template corresponding to the Capability Class is set up. When a new Capability Class is generated, a new Capability Template is also created and registered to the library. If one MSU could use different activities, several Capability Templates are selected. By filling the adequate Capability Template, the Capability Profile for the MSU is generated. The generated Capability Profile is registered in the Capability Profile library.

The flow shown on the right of Fig. 4 shows the development processes for a manufacturing application. The development process is roughly divided into the requirement analysis process and the MSU selection process. The requirements analysis process is to analyze the requirements and to describe the requirements as required by the Capability Profiles. First, a suitable manufacturing domain is selected. Second, requirements are decomposed into several primitive requirements by reference to the activity tree in the domain and activities are then appointed from the primitive requirements. Third, the Capability Class is adopted. Forth, the adequate Capability Template corresponding to the Capability Class is set up. The Capability Template is filled with specific requirements in order to generate a required Capability Profile. Next is the MSU selection and verification process. A MSU selection process starts with a required Capability Profile for a given activity. A desired set of MSUs has a corresponding set of Capability Profiles that match the capabilities required for a given activity. For each required Capability Profile, a search for matching Capability Profiles that represent available MSUs is performed. When a match exists, the MSU is added to a list of candidates. When a match does not exist, a new MSU is developed to meet the required Capability Profile, the required Capability Profile is decomposed into a combination of several Capability Profiles, or requirements are reconsidered against existing profiles. The selected MSUs are verified against the manufacturing application requirements according to interoperability criteria. Manufacturing system configurators or developers develop the manufacturing application by combining MSUs in the candidate list.

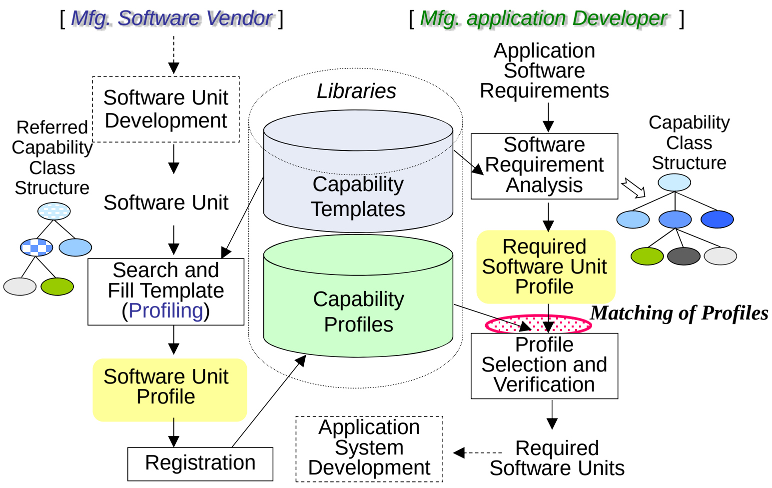

Fig. 4. Basic services using capability profiles.

4 Extended Services Provided by Using Capability Profiling

Using the capability profiling methodology provided by ISO 16100, new extended interoperability services are proposed.

4.1 Development of the Manufacturing Application by Integration of Manufacturing Software Units

ISO 16100 provides a standardized method to describe capabilities of manufacturing software in terms of the MSU Capability Profile. To use this, the manufacturing software capability catalogue can be organized. The manufacturing software capability catalogue is constructed by the Capability Template libraries and the Capability Profile libraries. The Capability Templates are registered with the identifier of the MDM. Probably, each vendor or vender group has their own MDM. The Capability Templates are categorized by MDM. Each library corresponds to a MDM. Each Capability Profile has an identifier of the Capability Template used. As a result, Capability Profiles are also categorized by MDM. MSUs themselves are in the corresponding vender's database. When the manufacturing software capability catalogue is open, the system configurator or developer use the catalogue to find and select an adequate MSUs, and develop the manufacturing application by integrating MSUs which are downloaded from each vendor's database.

Fig. 5 shows the case where the manufacturing software capability catalogue is in the cloud environment. The function for registration of the Capability Template and Capability Profile, the function for selection of an adequate Capability Template to describe requirements, and the function for matching the required Capability Profile with MSU's Capability Profiles are also provided as cloud services.

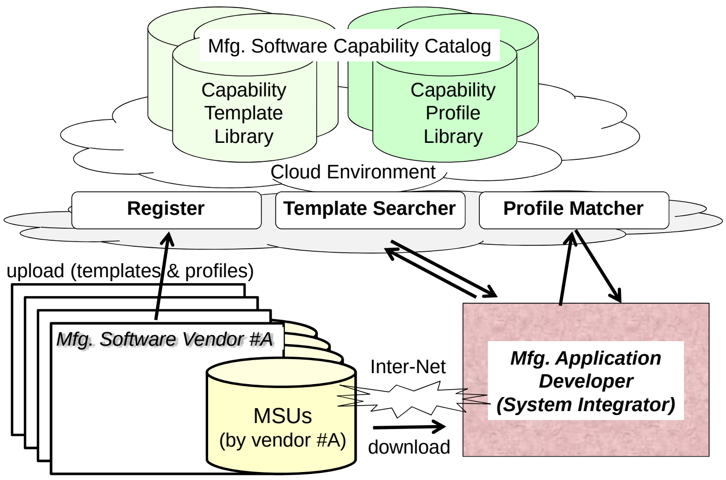

Fig. 5. Development of the manufacturing application using the cloud environment.

4.2 Extended Information Exchange

ISO 16100 also provides the way to exchange MSU's capability as information through a Capability Profile. This enhances interoperability services. On the dynamic configuration of manufacturing application, adequate MSU can be invoked when its capability is required through matching of Capability Profiles. The system configurator has required Capability Profiles, and organizes the manufacturing application by plug-and-play of MSUs which have the Capability Profile matched with the required profile. The MSU itself could become the configurator. In this case, the application is autonomously organized by direct communication among MSUs and plug-and-play of MSUs. The Capability Profile Library for searching adequate

MSUs is connected through the Capability Profile Interface (CPI). An MSU can have its own Capability Profile inside itself. In this case, the MSU itself has an interoperability function. Fig. 6 shows a configuration of the manufacturing application using plug-and-play of MSUs

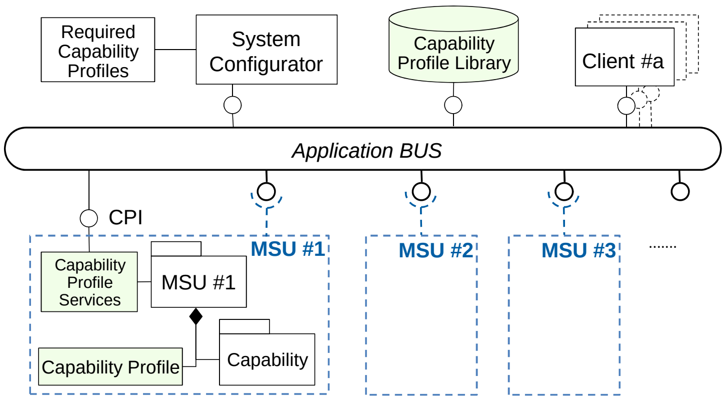

Fig. 6. Configuration of the manufacturing application by plug-and-play of MSUs.

5 Extension of Methodology for Manufacturing Software Capability Profiling

Extensions of the capability profiling methodology provided by ISO 16100 are required for implementation of proposed interoperability services. Some of required extensions are introduced.

5.1 Integration of Local Manufacturing Domains

For the matching procedure of two Capability Profiles, the MDDs has an important role. When considering the enhancement of interoperability services as mentioned in Chapter 4, integration of local MDD sets are required. One solution is proposed below.

In every MDM, one unique set of MDDs is available. In a Capability Template in terms of the Capability Class, the capabilities of required and existing MSUs are described using MDDs. An adequate Capability Template is selected corresponding to the activity which an MSU enables or supports. If there are several MSUs which enable the same activity, each profile is generated using the same Capability Template. A company and/or a company group usually has their own MDM and set of MDDs. If a manufacturing software vendor wants to widely distribute his developed MSU, the vendor makes several profiles for one MSU corresponding to each MDM. When naturally considering a manufacturing domain, the manufacturing domain corresponds to a manufacturing phase such as machining, assembly and diagnosis. Fig. 7 shows the above mentioned consideration.

When extending the manufacturing domain to this manufacturing phase, the MSU's interoperability is also extended. To extend the manufacturing domain concept, sets of MDDs for each company group should be integrated to make one MDM corresponding to a manufacturing phase. As one of solution, a MDD dictionary is introduced.

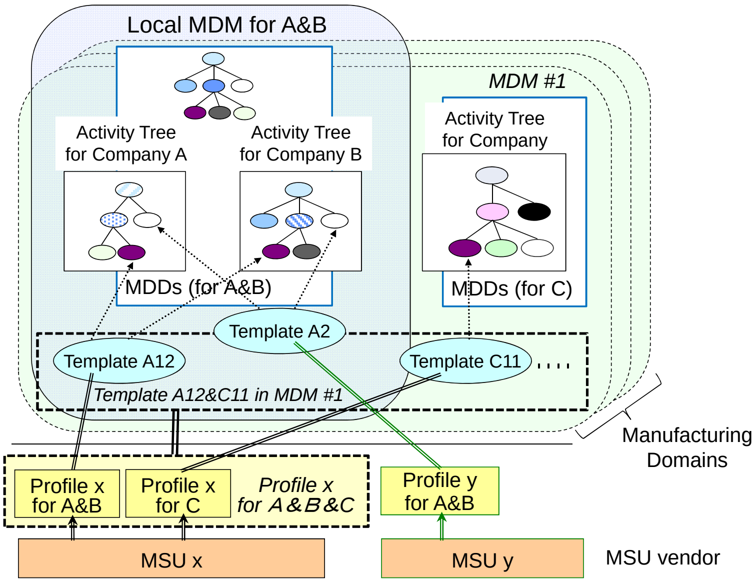

Fig. 7. Extension of MDM concept and integration of MDDs.

5.2 MDD Dictionary as An Domain Integration Tool

Items in the MDD Dictionary can be MDD classes, attributes, values of attributes, units of measure, qualifiers of measure, currency, data types, representations such as definitions, terms, abbreviations, images and languages, and other things that need to be referenced to within the Capability Profiles and Capability Templates. Here the ISO 22745 methodology [9] is applicable to construct the MDD Dictionary as shown in Fig. 8. In order to be compatible with ISO 22745, the Identification Schema in eOTD shown at the bottom left of Fig. 8, is used. ISO 22745 specifies a system of descriptive technology for representing, handling and exchanging master data. The registration authority identifier (RAI), data identifier (DI) and version identifier (VI) constitute the global unique identifier. Items in the MDD Dictionary can be created in two ways. The first one is that items are standardized by authorities as the data sources. The second one is that items are informatively copied from other local dictionaries which would be the sources of these items. Here, other local dictionaries include sets of MDDs in some MDMs and some other standards. Each item in the MDD Dictionary has one or more definitions, one or more terms or abbreviations and corresponding sources. For each such source, the link would be added to the item record in the MDD Dictionary. The MDD Dictionary serves as a 'Bridge' among different local dictionaries [8].

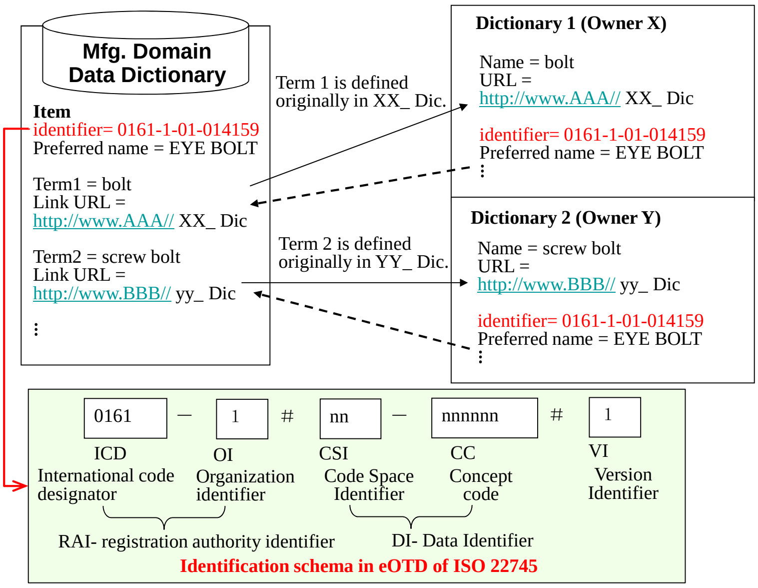

Fig. 8. The Domain Data Dictionary as a "Bridge" among different local manufacturing domains [8].

6 Conclusions

New manufacturing application development methods using the manufacturing software capability catalogue as a cloud service and a new manufacturing system configuration method using plug-and-play of manufacturing software are newly proposed. To enhance such interoperability services, extensions and intensions of ISO 16100 are required. For example, there are work items such as standardization of the construction method for MDM and MDDs, standardization of interface service including CPI, standardization for a manufacturing software capability catalogue such as contents and format, and standardization of service access to a Capability Profile and Capability Template. Now, the new project: ISO 16300 series is planed for these extensions. Furthermore, the implementation of ISO 16100 methodology would be expected to provide useful experience and help to define future directions.

Acknowledgments. The author thanks ISO/TC 184/SC 5/ WG 4 members and MESX joint WG in MSTC members for fruitful discussions and their useful effort to complete international standards. The author is also grateful to Dr. U. Graefe, the previous convenor of ISO/TC 184/SC 5/ WG 4 for his helpful assistance with the writing of this paper in English.

References

1. ISO 16100-1:2009 Manufacturing software capability profiling for interoperability Part 1: Framework. (2009)
2. ISO 16100-2:2003 Manufacturing software capability profiling for interoperability Part 2: Profiling methodology. (2003)
3. ISO 16100-3:2005 Manufacturing software capability profiling for interoperability Part 3: Interface services, protocols and capability templates. (2005)
4. ISO 16100-4:2006 Manufacturing software capability profiling for interoperability Part 4: Conformance test methods, criteria and reports. (2006)
5. ISO 16100-5:2009 Manufacturing software capability profiling for interoperability Part 5: Methodology for profile matching using multiple capability class. (2009)
6. ISO 16100-6:2011 Manufacturing software capability profiling for interoperability Part 6: Interface services, protocols for matching profiles based on multiple capability class structure. (2010)
7. Matsuda, M., Wang Q.: Software Interoperability Tools: Standardized Capability-Profiling Methodology ISO 16100. Enterprise Architecture, Integration and Interoperability, IFIP Advance in information and Communication technology Series, vol. 326, pp. 140--151, (2010)
8. Wang Q., Matsuda, M.: Manufacturing application interoperability using software capability catalogue. Enterprise Interoperability V: Shaping Enterprise Interoperability in the Future Internet , IFIP WG5.8, Springer, pp. 141--152 (2012)
9. Radack, G. ISO 22745: The Standard for Master Data 28 Nov, 2009. http://findarticles.com/p/articles/mi_qa3766/is_200710/ai_n27997247/