INTERNATIONAL STANDARD

ISO
13281

Industrial automation systems — Manufacturing Automation Programming Environment (MAPLE) — Functional architecture

Systèmes d'automatisation industrielle — Environnement de programmation pour l'automatisation industrielle (MAPLE) — Architecture fonctionnelle

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.

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.

International Standard ISO 13281 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 5, *Architecture and communications*.

Annexes A to C of this International Standard are for information only.

Introduction

Over the recent past, manufacturing systems have become considerably more flexible and have acquired greater functionality. The numbers and types of component devices of manufacturing systems, such as NC machines, robots, automated guided vehicles and programmable controllers, have increased. Furthermore, there is a definite trend for some of these devices to be incorporated in manufacturing cells. Manufacturing engineers are thus required to develop and update programs not only for many kinds of individual devices but also for combinations of devices and ultimately manufacturing cells. Due to this fact, the difficulty of integrating and programming the control of manufacturing operations has increased.

Manufacturing programs have an intense need for a large variety of manufacturing data, including product oriented data, process oriented data, operation oriented data and management oriented data. This diversity means that manufacturing data has a much more complicated and varied schema than the usual processing data encountered in other systems, e.g., business systems. Therefore, the use and management of manufacturing databases requires a special high-level technology.

MAPLE addresses the following problems that have been traditionally recognized to be within the domain of manufacturing application programming languages for automated production:

— Manufacturing presents a diversity of tasks with widely varying requirements and constraints. Often, addressing these tasks requires programming. Because of the diversity of requirements and constraints, a variety of manufacturing application programming languages have been found necessary.

— Typically, each manufacturing application programming language has its own unique environment of development methodologies, development, debug and simulation tools, and run-time services. Because these environments are stand-alone, it is difficult to achieve convenient access to the manufacturing databases.

— As a result, it is only with difficulty that an application developer or designer may coordinate the use of differing manufacturing languages for the individual tasks of a complete project, though this is a common need.

— Similarly, it is with great difficulty that systems engineers and integrators combine programs developed using different manufacturing languages, because they use or require different run-time services.

To address these problems, a language-independent manufacturing automation programming environment (MAPLE) is being standardized. This International Standard represents the functional architecture of MAPLE as a first step towards achieving such an environment. MAPLE is a structured set of capabilities that connects the objects such as data used in advanced manufacturing technologies to the required user oriented tools.

This International Standard for the MAPLE functional architecture specifies the functionality and interconnection of the components in the environment. It is intended for the technical committees, subcommittees and working groups of those standardization bodies whose mandate will be to develop the standards for the MAPLE services and interfaces, as well as for the commercial developers of MAPLE.

The MAPLE environment can replace existing in-house solutions that have been created by system integrators during the last decade to solve the above mentioned problems.

This support facility for programming will need a set of functionalities that are typical for the manufacturing environment, mainly because of the specific requirements of programming automated devices. Environments to support the programming of other automated devices outside the manufacturing domain (e.g. a transportation system with unmanned trains) will have similar architectures to MAPLE. Nevertheless, the functionality and implementation of the components needed by these other environments will be different from MAPLE.

MAPLE will provide the following benefits, which will lead to considerable time and cost reduction:

– easy and quick development of manufacturing programs;

– easy and quick updating of manufacturing programs;

– unified access for distributed manufacturing databases;

– unified management of manufacturing databases;

– effective utilization of manufacturing software tools;

– provision of a framework for future manufacturing software tools and data models.

Industrial automation systems — Manufacturing Automation Programming Environment (MAPLE) — Functional architecture

1 Scope

This International Standard specifies the functional architecture of MAPLE, a Manufacturing Automation Programming Environment. MAPLE is a common vendor-independent neutral support facility for the programming of multiple manufacturing devices and controls. Thus, MAPLE offers a single environment for the programming of a number of devices and controls not necessarily made by the same company. In addition, MAPLE does not require specifying specific devices and controls at programming time.

To the extent that it is concerned with the programming of manufacturing devices and controls, MAPLE will support the following areas:

- connections between various manufacturing data and manufacturing application programs;

- management of several manufacturing databases;

- sharing of manufacturing application programs and manufacturing software tools.

The scope of this International Standard will be relevant to the following users of the MAPLE standard:

- developers of manufacturing application programs;
- operators editing manufacturing programs;
- engineers who need to refer to manufacturing data.

2 Normative reference

The following standard contains provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the edition indicated was valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent edition of the standard indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/TR 11065:1992, *Industrial automation glossary.*

3 Definitions

For the purposes of this International Standard, the definitions given in ISO/TR 11065 and the following definitions apply.

3.1 Data Translator: a Manufacturing Software Tool for converting the representation of data.

3.2 Dictionary Manager: a Manufacturing Software Tool to facilitate the manipulation of the Manufacturing Data Dictionary and the Manufacturing Software Tool Dictionary.

3.3 Execution Manager: a software tool that controls the sequence of execution of internal processes of MAPLE and the external Manufacturing Application Programs and Manufacturing Software Tools.

3.4 Manufacturing Application Program: manufacturing software which has command and information connections with MAPLE (e.g. CAD systems), but is not registered in the Manufacturing Software Tools Dictionary.

3.5 Manufacturing Database: a data repository, possibly distributed, containing product, process, facility and management oriented data.

3.6 Manufacturing Data Dictionary: a collection of data schema (meta-data) describing data in the Manufacturing Database.

3.7 Manufacturing Data Manager: a Manufacturing Software Tool enabling access to the Manufacturing Databases.

3.8 Manufacturing Software Tool: a software program registered in the Manufacturing Software Tool Dictionary, which provides simple or complex, manufacturing related functionality for the support of Manufacturing Application Programs. (e.g. simulators, editors).

3.9 Manufacturing Software Tool Dictionary: a collection of descriptions of the capabilities of Manufacturing Software Tools and Manufacturing Application Programs.

3.10 MAPLE: a Manufacturing Automation Programming Environment, a common vendor-independent neutral support facility for the programming of multiple manufacturing devices and controls.

3.11 MAPLE Engine: part of the standardized core of MAPLE, a software tool for receiving and handling requests to MAPLE and the initialization of the entire environment.

3.12 Software Tool Linker: a Manufacturing Software Tool to select and sequence other Manufacturing Software Tools and Manufacturing Application Programs to fulfil MAPLE Engine requests.

4 MAPLE functional architecture and its components

The MAPLE functional architecture, its components and interfaces between these components, and the interfaces to the outside world are shown in figure 1. To facilitate the description of the components, and (in clause 5) the interfaces, the components and interfaces shown in figure 1 are labelled with the appropriate section numbers describing them. Figure 2 shows the information and control flow between the MAPLE components themselves and the outside world. The following provides more functional detail on the architecture and components of the Manufacturing Automation Programming Environment.

The functional architecture of this environment consists of the the MAPLE Engine, the Manufacturing Data Dictionary, the Manufacturing Software Tool Dictionary, a Dictionary Manager, a Manufacturing Data Manager, an Execution Manager and a Software Tool Linker, which provide standardized functionality and interfaces to the Manufacturing Application Programs, the Data Translator, the Manufacturing Software Tools and the Manufacturing Databases. The user interface is through the Manufacturing Application Programs and the Manufacturing Software Tools.

The MAPLE Engine is the driver of the whole environment. To keep track internally of the variety of types of data being used in the Manufacturing Data Bases, a Manufacturing Data Dictionary, describing standard manufacturing data models, is provided. Similarly, to facilitate the use of Manufacturing Software Tools and Applications connected to MAPLE, especially if a number of such Tools and Applications have to be linked to accomplish a given task, a Manufacturing Software Tool Dictionary, describing the Tools' functionalities as well as their input and output requirements, is provided. The actual data handling is achieved through MAPLE's Manufacturing Data Manager.

4.1 MAPLE Engine

The MAPLE Engine provides the following functionalities:

— allow for the initialization of the entire environment;

— provide the MAPLE Interface between the MAPLE and the outside world;

— accept external requests for services of the environment, related to the Software Tool Linker, Manufacturing Data Manager, Data Translator and Dictionary Manager or any other Software Tool or Application Program connected to MAPLE;

— interpret and process external requests by calling on either the Software Tool Linker or the Execution Manager directly;

— provide status information to the user on request through a Manufacturing Application Program or a Manufacturing Software Tool;

— pass commands coming from the Execution Manager and directed to an external Manufacturing Software Tool or Application Program through the MAPLE Interface;
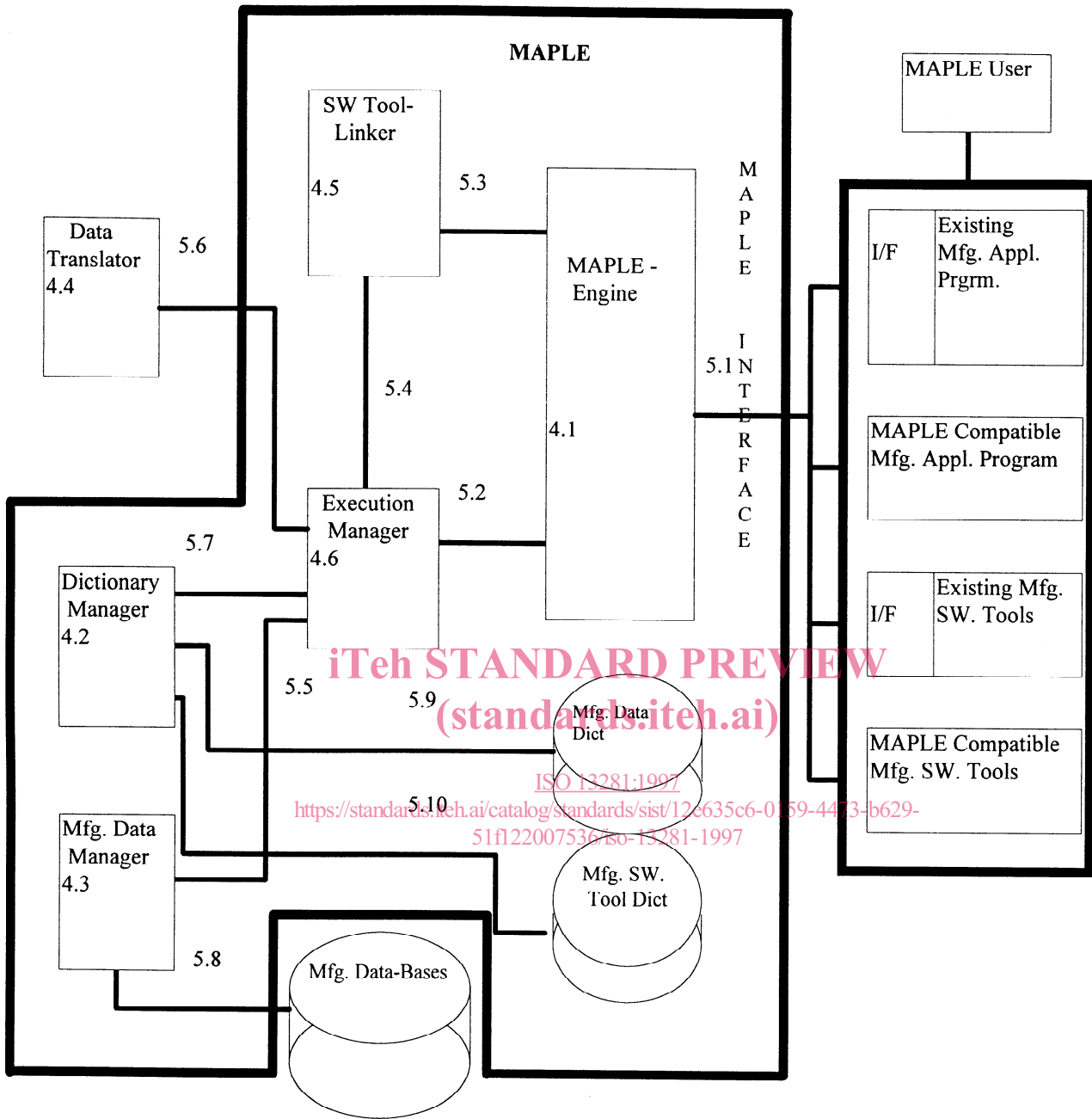
Figure 1 - MAPLE functional architecture and interfaces

–   pass data between the Execution Manager and external Manufacturing Software Tools or Application Programs through the MAPLE Interface.

The MAPLE Engine is responsible for the handling and the control of external requests to MAPLE. For this purpose the MAPLE Engine provides an interface to all user applications and software tools connected to MAPLE. The external requests to MAPLE are interpreted by the MAPLE Engine, which decides which actions should be taken. If the external request can be handled by one single action of a software tool or application (simple request) then a request for this action is passed to the Execution Manager who will take care of the execution. If the external request involves more complex actions, the MAPLE Engine issues a request to the Software Tool Linker to create a task list of actions that will fulfil the original request. The execution of the actions appearing in the task list is managed by the Execution Manager. In reply to the original request issued to MAPLE, the MAPLE Engine will provide the requested information, data or action as well as status information on the success or failure of handling the particular request made.

4.2   Dictionary Manager

The Dictionary Manager permits manipulation of the Manufacturing Data Dictionary and the Manufacturing Software Tool Dictionary through the provision of the following functions for either one of the two dictionaries:

–   adding an entry to the dictionary;

–   deleting an entry from the dictionary;

–   editing an item in the dictionary;

–   searching for and retrieving a dictionary entry.

All activities involving either the Manufacturing Data Dictionary or the Manufacturing Software Tool Dictionary are solely handled by the Dictionary Manager. These activities consist of adding, deleting, editing and search for and retrieving of entries in either one of these two dictionaries. The requests for such activities, including any associated data, come to the Dictionary Manager from the Execution Manager, triggered by requests from a user; the Manufacturing Data Manager, which may require information regarding a Manufacturing Database access; the external Data Translator, which requires format information for one of its input or output files; or the Software Tool Linker, which requires information about the capabilities and needs of Manufacturing Software Tools, and the required formats of inputs and outputs of these tools. All these requests are controlled by the Execution Manager, who is in fact the process invoking the Dictionary Manager and taking care of data passing from the requestor to the Dictionary Manager and vice versa. The interfaces between the Dictionary Manager and the Manufacturing Data Dictionary and the Manufacturing Software Tool Dictionary are described in 5.10.

4.3   Manufacturing Data Manager

The functionalities of the Manufacturing Data Manager can be summarized as follows:

– access data in the Manufacturing Databases (select, insert, delete, update);

– provide information concerning Manufacturing Databases;

– control user access to the Manufacturing Databases (security);

– control database integrity;

– allow maintenance of the Manufacturing Databases (e.g. create database)

The programming and control of the different component devices of manufacturing systems requires an extensive amount of manufacturing data. This data, including product oriented data, production oriented data, operation oriented data and management oriented data, tends to have a very complex data structure. As a result of this, actual manufacturing databases in most companies tend to have also a complex physical structure (e.g. distributed databases) requiring high level and special technology for the transparent use and management of this data.

The Manufacturing Data Manager provides the means of access to the Manufacturing Databases in response to requests from the Execution Manager. Requests for access to data in the Manufacturing Databases can come from the MAPLE Engine, the Software Tool Linker and the Data Translator. It accesses any required information from the Manufacturing Data Dictionary via a request to the Execution Manager. It includes the regular functions of a Database Manager such as user access control and security as well as maintenance of database integrity. It also provides information concerning the Manufacturing Databases. MAPLE and the applications relying on it, performs all its Manufacturing Database accesses (select, insert, delete, update) through the Manufacturing Data Manager.

4.4   (External) Data Translator

The Data Translator is a software tool external to the MAPLE environment, invoked by the Execution Manager and translates data from one specified data model to another. In order to perform this task, the Data Translator accesses both the input and output data models stored in the Manufacturing Data Dictionary through requests to the Execution Manager.

Typically the Software Tool Linker will have identified the need for translation and will invoke the Data Translator through the Execution Manager to perform a particular data translation task. The formats to be used within the data translation mechanism are held within the Manufacturing Data Dictionary, hence there is a bi-directional communication initiated between the Data Translator and the Dictionary Manager via the Execution Manager, to obtain the necessary format information. This activity is controlled by the Execution Manager. The next stage of the Data Translator phase is to read the data which is to be translated and to write the translated output. These data sets can be residing at intermediate storage spaces controlled by the Execution Manager or can be in the Manufacturing Databases. In the latter case the Manufacturing Data Manager is invoked.

The method utilized for data translation is considered an implementation issue. The Software Tool Linker passes the two data formats and the Data Translator implementor may select any means to achieve the required translation (i.e. single phase [source → destination], double phase [source → neutral format, neutral format → destination] are valid).

A special case of the Data Translator is a data filter, a Manufacturing Software Tool which performs selective data translation.

4.5 Software Tool Linker

The Software Tool Linker responds to requests from the MAPLE Engine to select and sequence other Manufacturing Software Tools. The Software Tool Linker, a special manufacturing software tool itself, creates one virtual manufacturing software tool from multiple manufacturing software tools. Using this special manufacturing software tool, users can implement 'new' manufacturing software tools, by combining manufacturing software tools which have already been developed and catalogued within the Manufacturing Software Tool Dictionary.

The Software Tool Linker provides the following functionalities:

— interpret MAPLE Engine requests to determine requested functionality;

— examine through the use of the Dictionary Manager, the Manufacturing Software Tool Dictionary to identify the Manufacturing Software Tools required to achieve the requested functionality;

— provide to the Execution Manager a task list of subsequent actions to be taken to achieve the original request of the MAPLE Engine. A task description will typically contain the name of the tool to be used, the commands to be given to this tool, the location of the input data and the location where to provide the output data;

— provide status information on the linking process to the MAPLE Engine.

4.6 Execution Manager

The Execution Manager is responsible for the execution of tasks issued by the MAPLE Engine or task lists issued by the Software Tool Linker. Its functions are :

— accept tasks from the internal MAPLE components or task lists from the Software Tool Linker;

— for each single task requested by an internal MAPLE component issue control commands to the Manufacturing Software Tool selected, in order to achieve the requested functionality;

— for task lists requested by the Software Tool Linker, issue control commands in the proper sequence to the Manufacturing Software Tools selected, in order to achieve the requested functionality. In the case of Software Tools invoked external to MAPLE, the control commands are transferred through the MAPLE Engine and the MAPLE Interface;

— accept status information from the tools running. In the case of Software Tools external to MAPLE, the status information is transferred through the MAPLE Interface and the MAPLE Engine;

— provide status information to the requesting MAPLE component or to the Software Tool Linker concerning the execution of respectively a task or a task list;

— set aside any intermediate storage space required for data to be exchanged between successive Manufacturing Software Tools. Intermediate storage for data and associated data models can be assigned either internally to the Execution Manager, or in the Manufacturing Databases and Manufacturing Data Dictionary, respectively;

— manage the data flows between Manufacturing Software Tools and intermediate storage space provided by the Execution Manager. In the case of Software Tools external to MAPLE, the data is transferred through the MAPLE Engine and the MAPLE Interface;

— keep track of the stack of tasks to be executed, following an original external request. Note that the original task can lead to several sub-tasks, to be executed before the original task can be completed. For example, a single task to call the Data Translator can lead to several sub-tasks issued by the Data Translator to receive information from the Dictionary Manager.

4.7 Manufacturing Data Dictionary

The Manufacturing Data Dictionary is basically a facility (database) to store a machine readable collection of descriptions of data schema. A description is provided for data in the Manufacturing Databases, as well as for temporary data that the Execution Manager stores as intermediate data between a set of Manufacturing Software Tools that are concatenated to achieve a desired functionality. The content of the Manufacturing Data Dictionary provides instructions on how to read and interpret the data it describes. An entry in this dictionary has two main aspects: format of the data and meaning of the data. The format of the data is information that is ultimately needed by the Manufacturing Data Manager, the meaning of the data is information needed by the Data Translator, the Software Tool Linker and other Manufacturing Software Tools and Manufacturing Application Programs. The Manufacturing Data Dictionary is a database with a main table, in which each record represents one manufacturing data entity, and other related tables needed to describe table structures and relations between tables of the manufacturing database.

Manipulation of the Manufacturing Data Dictionary is achieved through the interface with the Dictionary Manager. Essentially the Dictionary Manager reads, writes or edits the dictionary as required, to achieve the goal of add, delete, edit or search and retrieve commands. The response of the Manufacturing Data Dictionary consists of the requested dictionary content information, as well as any status information on the dictionary manipulation.