NISTIR 5992

Analysis of AP213 for Usage as a Process Plan Exchange Format

Michael Iuliano
Albert Jones
Shaw Feng

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards and Technology
Gaithersburg, MD 20899-0001

March 1997

NIST

NISTIR 5992

Analysis of AP213 for Usage as a Process Plan Exchange Format

Michael Iuliano
Albert Jones
Shaw Feng

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Gaithersburg, MD 20899-0001

March 1997

U.S. DEPARTMENT OF COMMERCE
William M. Daley, Secretary

TECHNOLOGY ADMINISTRATION
Mary L. Good, Under Secretary for Technology

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Arati Prabhakar, Director

NIST IR 5992

Analysis of AP213 for Usage as a Process Plan Exchange Format

Michael Iuliano, Albert Jones, and Shaw Feng
Manufacturing Systems Integration Division
Manufacturing Engineering Lab
National Institute of Standards and Technology, Gaithersburg MD, USA

Abstract

The objective of the Computer-Aided Manufacturing Engineering (CAME) project is to identify generic interfaces which can be used to demonstrate the integration of manufacturing engineering software applications. A number of software tool kits are envisioned to meet that objective. The first of these tool kits is called the Manufacturing Engineering Tool Kit (METK). It focuses on the integration of operations planning and Numerical Control (NC) program verification. The METK strategy is to use a process plan as a means of transferring the required electronic data between these applications. From the METK viewpoint, a process plan is a collection of work elements which describe the tasks to be done and the resources needed to do them. The exchange of these plans between these packages is the focus of this paper. The paper describes the commercial applications which make up the METK and our efforts to use the protocol specified in ISO DIS 10303-AP213 to exchange plans among these applications.

Keywords: integration, manufacturing engineering, NC programs, process plans, simulation

1. Introduction

There are typically three major phases in the product realization process: design, engineering, and production. During the design phase, customer requirements are used to generate a product design. This design describes **what** will be produced. Typically, it will include a product model and bill of materials. During the engineering phase, this design is used to generate a process plan. This plan provides a recipe for **how** to make the product described in the design. It includes the processes to be used, the machines on which these processes will be implemented, the order in which machines will be visited, and all other required materials and resources. During the production phase, this process plan is used to generate a schedule for the actual fabrication, inspection, and shipment of the physical product. While a large number of software applications exist to carry out many of the functions within these phases, they are still, for the most part, human intensive activities. Humans are still required to 1) translate and transfer data between these applications, 2) execute the applications, perhaps many times, 3) interpret the results, and 4) make decisions based on those results.

Research programs in every industrialized nation in the world have focused on the automation and integration of these activities. These programs have familiar names - CIM (Computer Integrated Manufacturing), EI (Enterprise Integration), AM (Agile Manufacturing) and VM (Virtual Manufacturing). The National Institute of Standards and Technology (NIST) continues to play a major role in the development and testing of standards to support these programs. One of its current projects - Computer-Aided Manufacturing Engineering (CAME) - is aimed at the development and demonstration of standards to enable the integration of software applications used during the engineering phase [1]. A Manufacturing

Engineering Tool Kit (METK) is being developed as part of CAME. METK focuses on the integration of two of those applications: operations planning and Numerical Control (NC) program verification. This paper describes attempts to use the Draft International Standard, ISO DIS 10303-AP213 [2], to exchange information between these two applications.

The paper is organized as follows. Section 2 describes the applications which make up the METK, gives a high level description of a process plan, and provides an example of a plan being used in the METK. Section 3 summarizes ISO 10303-AP213. Section 4 provides a detailed explanation of the mapping developed between the example process plan given in section 2 and AP213. Section 5 summarizes the problems we expect to encounter in using that mapping.

2. A Process Plan for the METK

2.1 A Systems Overview of the METK

A high-level system diagram for the METK is shown in Figure 1. The software application which implements the Product Data Management (PDM) function is called MATRIX®, from ADRA Systems*. MATRIX® manages an object-oriented database of distributed files, the applications that create those files, and the business process that governs their life cycle. This business process is defined in a workflow management scheme. The Computer-Aided Design application is Pro/ENGINEER® (Pro/E), from Parametric Technology Corporation. Pro/E is used to create a product design file. This design includes a solid model representation of the final geometry of the product, and a part blank which represents the initial geometry. This design file can be stored as a ProE neutral file or AP203 file [3]. It is retrieved by the operations planning application, ICEM PART®, from Teknomatics. ICEM PART® uses an automated feature recognition algorithm to detect machineable features from the geometry model. It then allows the user to select the machine, the tool set, and the required jigs/fixtures. ICEM PART® uses the derived feature definitions and the user-defined selections to create the necessary operations sheets, tool paths and NC programs. The operations sheets, fixture and tool list, and NC program are used to drive the NC verification application, VNC®, from Deneb Robotics. (Developing a standard representation for this collection of information is a major focus of METK and this paper). VNC® uses simulation models of the machine tool and its controller, fixtures, cutting tools, and part blank to generate and visualize the output of the NC program. It can check for simple types of errors, such as missing tools, and also detect collisions between the tool and the part, fixture, or table.

Currently, all METK applications reside and execute on a single Silicon Graphics ONYX workstation running the IRIX5.3® operating system. This workstation is in the Advanced Manufacturing Systems and Networking Testbed (AMSANT) facility at NIST. AMSANT was established to support testing of high performance computing and networking hardware for next generation manufacturing systems.

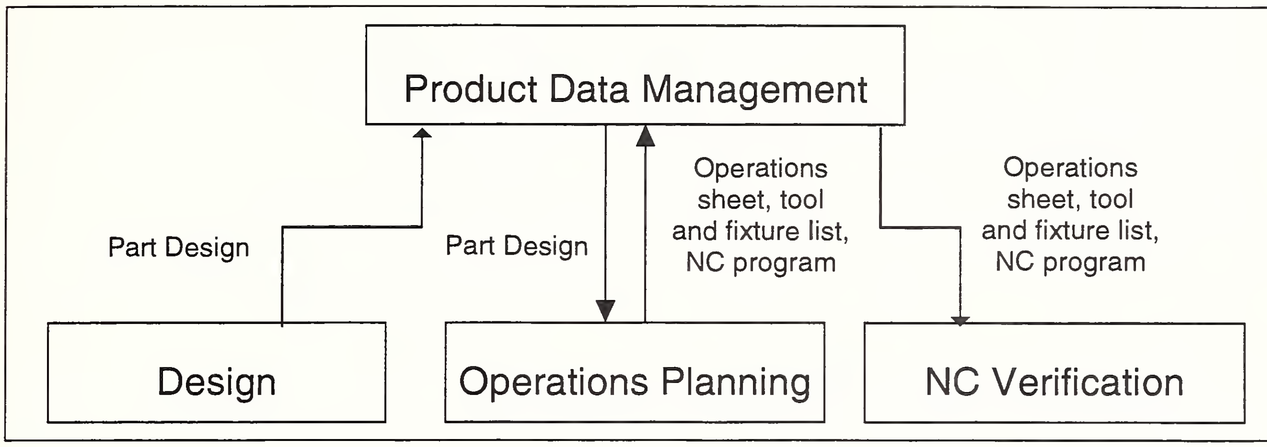

Figure 1. METK Functional Overview

2.2 Integration using Process Plans

As indicated in Figure 1, the integration of the operations planning and NC verification application is coordinated through the product data management application. The METK is using the concept of a process plan [4] to facilitate that integration. As discussed in [4], a process plan can be viewed as a recipe for **how** to make the product described in the design. Currently, it contains three sections: Header, Resources, and Procedures.

- The Header section contains all administrative information about the plan. It may include a plan identifier, version number, revision date, part name, part identifier, planner's name, etc.
- The Resource section lists all of the resource files needed to implement the plan including tools, fixtures, materials, machine, software programs and, possibly, other process plans.
- The Procedure section identifies all of the steps necessary to simulate the actual execution of the plan. Example steps include LOAD_TOOL, LOAD_FIXTURE, and RUN_NC_PROGRAM. Each step has a step number, work description with associated requirements, and one or more precedence constraints. These constraints indicate the order in which the steps can be executed (if no constraints are given, the plan is executed in the order given by the step numbers).

HEADER Section
plan_id=P12345
part_name=Air_frame_test_part
creation_date=10/24/96
planner=Mike Iuliano

RESOURCE Section
machine_id=CINC_MILA_T30
tool_name=1/4" TWIST_DRILL
tool_name=1/2" CENTER_DRILL
tool_name=1/8" BALL_NOSE_END_MILL
tool_name=SHANK_END_MILL
fixture_name=vise
workpiece_name=Air_frame_blank
nc_program=Air_frame.cnc

PROCEDURE Section
Step 1 LOAD_MACHINE
machine_id = CINC_MILA_T30
machine_controller = GE2000
end_step
Step 2 LOAD_TOOL
tool-name = TWIST_DRILL
tool_id = T266
magazine_slot = 1
end_step
Step 3 LOAD_TOOL
tool-name = CENTER_DRILL
tool_id = T271
magazine_slot= 2
end_step
Step 4 LOAD_TOOL

```
    tool-name =BALL_NOSE_END_MILL
    tool_id = T268
    magazine_slot = 3
    end_step
Step 5  LOAD_TOOL
    tool-name =SHANK_END_MILL
    tool_id = T234
    magazine_slot= 4
    end_step
Step 6  LOAD_FIXTURE
    fixture_name = vise
    fixture_id = V178
    ref_frame = x_axis
    x,y,z_offset = 152.4, 101.6, 44.45
    units = inches
    end_step
Step 7  LOAD_WORKPIECE
    workpiece_name = Air_frame_blank
    workpiece_id = W123
    ref_frame = fixture_name
    x, y,z_offset = 0, 0, 0
    units = inches
    end_step
Step 8  LOAD_NC_PROGRAM
    nc_program = Air_frame.cnc
    end_step
Step 9  RUN_NC_PROGRAM
    nc_program = Air_frame.cnc
    end_step
```

Figure 2. Example of a Simplified Process Plan

Figure 2 shows a simple instantiation of such a plan. The execution of this plan is then carried out by the NC verification application. That application must have a "front-end" which can 1) parse the plan, 2) retrieve (or build) geometry models and other program files, 3) set up the machine using steps 1 thru 8, and, finally, 4) simulate the NC program. The goal of the METK is to develop a generic representation and exchange protocol for such a process plan and its associated resource files. The benefit of such a representation and protocol is that this front-end will only have to be written once.

3. AP 213

This development effort commenced with an examination of the existing ISO 10303 suite of standards. ISO 10303, commonly referred to as STEP, is an International Standard for the exchange of product model data. STEP is organized as a collection of "Parts". There are six Part types: descriptive methods, integrated resources, application protocols, abstract test suites, conformance testing, and implementation methods. ISO 10303-213 is an Application Protocol (AP), which means it is used for archiving, sharing, an exchanging information. AP213, titled "Numerical Control Process Plans for Machined Parts", deals specifically with computer-readable NC process plans for machined parts. It defines the context, scope, and information requirements for representing a process plan. (As such, it was a potential candidate for the METK project). Finally, as with other APs, AP213 includes a collection of conformance requirements and test purposes which provide the basis for the abstract test suite which will be used to perform conformance testing of AP213 software implementations.

AP213 contains an information model that specifies the data elements (as well as the relationships between those elements) that make up a process plan. These data elements and their relationships are part of the Application Reference Model (ARM). The ARM is then mapped to an ISO 10303 resource model called an Application Interpreted Model (AIM). These models can be used to provide interfaces between the software systems that create various parts of the process plan. AP213 was designed to allow dissimilar systems, in particular Computer-Aided Process Planning (CAPP) and Computer-Aided Manufacturing (CAM) systems - to exchange the data found **within** a process plan. It was not intended to provide downstream manufacturing applications with the information they need to **execute** a process plan. In particular, it was not meant to provide an interface which could drive the execution of a process plan using a simulation - one of the principal requirements of the METK project.

4. A Mapping to AP 213

The process plan shown in Figure 2 contains a linearly ordered sequence of work elements and their associated value/attribute pairs. These work elements, together with the files listed in the resource section of the plan, provide the information needed to build and execute a simulation of a single machine manufacturing a part. They include: retrieve the geometry file for the machine; retrieve the geometry files for the required tools and place them in the assigned slots on the machine; retrieve the geometry model for the fixture and attach it to the machine at the designated coordinates; retrieve the geometry for the workpiece blank model and attach it to the fixture in the prescribed orientation; retrieve the NC program file and load it into the machine; and, finally, run the NC program.

To determine the potential for using AP 213 as a representation for such a plan, we carried out an extensive analysis of the AP 213 ARM information model and how it was constructed. Figure 3 depicts the basic data components of AP 213 ARM and the relations among them. This figure illustrates that, as far as AP213 is concerned, a process plan is composed of one or more activities which can be decomposed into sub activities. An activity is linked to the product definition data and is determined by the product shape. An activity describes the processing and resources required. Certain data is considered in-scope and certain data is considered out-of-scope.

Data which are considered inside the scope of AP 213 include: planning information contained in the NC process plans for machined parts; task instructions required to manufacture a part using numerical control; required NC programming information specified in the process plan; in-process inspection information specified in the process plan; shop floor information specified in the plan.

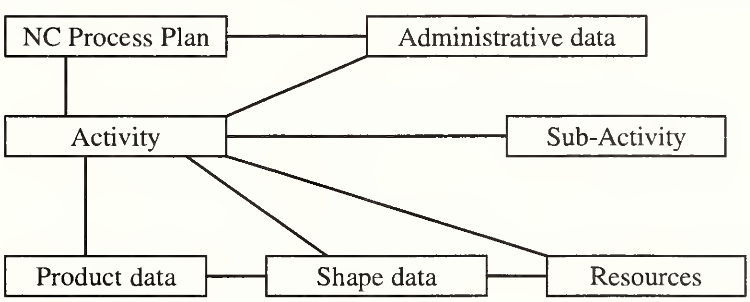

Figure 3: Simplified AP213 data model

Data which is considered outside the scope of AP 213 includes: preplanning NC process information, production planning, scheduling, continuous processes, make/buy analysis, costing, form features and drawings, operations planning, inspection planning, actual execution of the plan, and the NC source program. Since last two data items are critical to the implementation of the METK, we thought that we would have difficulty using the AP 213 ARM model directly. Therefore, we attempted to develop a mapping between process plan being considered for the METK project and the AP213 ARM model.

As noted above, the procedural steps necessary to execute the simplified process plan in Figure 2 include 1) loading the models for the machine, controller, tools and fixture and the workpiece; 2) loading the NC program; and 3) executing the NC program. Simply stated: set up the machine and run the program. If we examine the AP 213 ARM, we see that it supports activities called part handling, part loading, machine setup, and material removal. Each of these activities has several associated attributes. At first glance it might appear that a simple and straightforward mapping between the steps in the process plan in Figure 2 and these ARM activities could be developed. This, however, was not case.

The first problem was tooling data. In the METK process plan, tooling data is associated with the LOAD_ TOOL step in the procedure section which is, conceptually, part of machine setup. In the ARM model, however, tooling data corresponds to attributes associated with the MATERIAL_REMOVAL activity. Therefore, in the mapping shown in Table 1, the LOAD_TOOL step is associated with the MATERIAL_ REMOVAL activity and not the MACHINE_SETUP activity in the ARM.

The second problem involves the order in which parts and fixtures are loaded onto the machine. In the METK process plan shown in Figure 2, the fixture is loaded onto the machine first, and then the workpiece is placed in fixture. There is no direct analog to this in the ARM model. In AP213, there are two related activities: PART_FIXTURE_MOUNTING and PART_MACHINE_MOUNTING. There is, however, no FIXTURE_MACHINE_MOUNTING activity. There are three ways to use the AP 213 activities to address this problem. First, the fixture can be placed on the machine using the MACHINE_SETUP activity. The resulting fixture/machine combination would then be considered the "machine". The part would be placed onto this machine using the PART_MACHINE_MOUNTING activity (This scenario is depicted in the mapping table below). Second, the part could be placed in the fixture using the PART_FIXTURE_MOUNTING activity. The resulting part/fixture combination would then be considered the "part". This part would then be placed on the machine using the PART_ MACHINE_MOUNTING activity. Third, the fixture could be placed onto the machine using the PART_MACHINE_MOUNTING activity, with PART now understood to be the fixture. Then, the resulting fixture/machine combination would be viewed as "machine. The part would be inserted using the PART_MACHINE_ MOUNTING activity.

Table 1. Process Plan to AP213 Mapping

METK process plan entity	AP213 Entity
HEADER Section	
plan_id	NC_process_plan_version.id
part_name	Part_version.nomenclature*
creation_date	none
planner	Planning_group_member.name
PROCEDURE Section	
LOAD_MACHINE	Activity.description
machine_id	Machine.id
step number	Material_removal.number
LOAD_TOOL	MATERIAL_REMOVAL. activity
tool_name	Special_instruction.instruction_text*
tool_id	Tool_assembly.id
magazine_slot	Tool_data.tool_position
LOAD_FIXTURE	MACHINE_SETUP activity
fixture_name	Special_instruction.instruction_text*
fixture_id	Fixture_assembly.id
ref_frame	Special_instruction.type (one word)*
x,y,z offsets	Special_instruction.text *
LOAD_WORKPIECE	PART_MACHINE_MOUNTING activity

workpiece	Special_instruction.instruction_text*
workpiece_id	NC_part_name
ref_frame	Part_fixture_mounting.fixture_location_origin *
x,y,z offsets	Part_fixture_mounting.fixture_orientation *
LOAD_NC_PROGRAM	Activity.description
nc_program	Special_instruction.text
RUN_NC_PROGRAM	Activity.description
nc_program	Special_instruction.text

* - no direct mapping

From the preceding table, it is clear that several METK-defined procedures and attributes have no direct representation in AP 213. These procedures can only be mapped to the generic "Activity". The attributes are mapped to "Special_instruction.text". Additional limitations were also identified. They are listed below.

1. The date that an NC process plan is created is an important piece of information, and it should be recorded in process plan data. This piece of information is not defined in the ARM.

2. Tool names can not be represented in AP 213. Tool name is different than tool id. A tool name can be 1/4" end mill, 1/2" twist drill, 2" shank end mill, etc. There is no Tool_name attribute in AP 213. Tool name is used in CAPP systems. Without an attribute in AP213, the information can not be shared among operations planning, CAPP, and CAM systems.

3. There is no straightforward way to represent the mounting of a fixture on a machine tool. The position of a fixture relative to the machine origin and its orientation relative to the machine axes are critical to the verification of the NC program. An activity called FIXTURE_MACHINE_MOUNTING should be added to the ARM model with the attributes required to place an empty fixture on a machine table.

4. The name of actions, that is procedural steps, cannot be represented directly in AP 213. There is no attribute in the ARM entity called Activity to capture the activity's name. Examples of such names are "MILL_POCKET", "LOAD_NC_PROGRAM", "DRILL_HOLE", etc.

5. It is possible to support simple sequencing of activities, but arbitrary relationships between activities cannot be supported easily at this time.

5. Conclusions

We can draw three general conclusions from the preceding analysis. First, there are important philosophical differences regarding the concept of a process plan between AP213 and METK. In AP213, a process plan is viewed as a collection of the data needed to manufacture a part. Formal models are defined in AP213 to support the exchange of information between applications that **create** that data. In METK, a process plan is viewed as a recipe that includes both the data and the instructions needed to manufacture a part. Formal models are needed in METK to support the exchange of information between applications that generate process plans and applications that **execute** those plans. Second, in spite of these differences, a mapping can be defined so that the AP213 ARM model can support the execution of process plans with simple linearly ordered tasks like the one shown in Figure 2. The mapping is not simple, but it could be implemented in a custom interface. Finally, this interface would require substantial modifications to support process plans with more complicated precedence relations - such as AND/OR graphs, conditional branching, and parallelism - among the steps in the procedure specification.

* No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied.

Acknowledgments: The authors wish to thank Chuck McLean for his advice on process plan work elements. Work described in this paper was sponsored by the US Navy Manufacturing Technology Program and the NIST Systems Integration for Manufacturing Applications (SIMA) Program and is not subject to copyright.

References

1. McLean, C., “Computer-Aided Manufacturing Systems Engineering”, *IFIP Transactions: Advances in Production Management Systems*, North Holland, Amsterdam, Netherlands, 341-348, 1993.

2. ISO/DIS 10303-213, " Industrial automation systems and integration - Product Data Representation and Exchange - Part 213: Application Protocol: Numerical Control Process Plans for Machined Parts," International Organization for Standardization, 1 rue de Varambe, Case Postale 56, CH-1211 Geneva, Switzerland, 1996.

3. ISO/IS 10303-203, “Industrial automation systems and integration - Product Data Representation and Exchange - Part 203: Configuration Controlled 3D Design of Mechanical Parts and Assemblies," International Organization for Standardization, 1 rue de Varambe, Case Postale 56, CH-1211 Geneva, Switzerland, 1994.

4. McLean, C., “Interface Concepts for Plug-Compatible Production Management Systems”, *IFIP WG 5.7: Information Flow in Automated Manufacturing Systems,* North Holland, Amsterdam, Netherlands, 307-318, 1987.