Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2009)
"Energy and Technology for the Americas: Education, Innovation, Technology and Practice"
June 2-5, 2009, San Cristóbal, Venezuela.

Adding Part Features to a Computer-Assisted Manufacturing Process Planning System: A Continuing Case-study in University-Industry Cooperation

Manuel E. Bermudez
University of Florida, Gainesville, Florida, USA, manuel@cise.ufl.edu

ABSTRACT

We describe an on-going project at the University of Florida, sponsored by General Dynamics Inc. The goal of this multi-year project is to partially automate the creation of a process plan for manufacturing a "part" (in the past the M107 ballistic shell casing, more recently a pipe joint for oil exploration) at the General Dynamics manufacturing plants in Scranton and Red Lion, Pennsylvania, respectively. In the manufacturing process, the starting point is a steel billet. This undergoes a number of operations (cutting, foundry heating, initial molding, cooling, turning, ring application, additional cutting, nose and tail lathing, and finally sanding and painting) to produce a shell casing or a pipe joint. There are many choices for a machine on the plant floor (over a hundred) to perform any one of these operations. Thus, machine selection for the process plan is a major engineering design issue, involving process plan feasibility, machine tolerances, processing costs, waste of material, and time. In this paper, we describe the multi-year project we are carrying out at the University of Florida to partially automate this process planning activity. Four teams of 5-6 engineering students are involved, each team with a faculty coach, working together with personnel at General Dynamics and two external consultants. We describe the current project phase (year three), our tree-pruning algorithm for process plan selection and analysis, and this year's newly added component, the construction of a part via a Feature Library.

Keywords: Computer-assisted process planning, university-industry cooperation.

1. INTRODUCTION

With the advent of globalization, many US manufacturing companies have been forced to take steps to inrease the efficiency of their operations, in order to maintain their competitive edge. In some areas of manufacturing, the problem is compounded with an aging workforce, many of whom will soon retire and take valuable knowledge and experience with them. In light of this situation, General Dynamics, one of the largest contactors to the US Department of Defense, is taking steps to partially automate many of its manufacturing processes. Of special interest to their plants in Scranton and Red Lion, Pennsylvania, is the long, expensive process of engineering a process plan, a sequence of machine operations that transform a raw steel billet into the casing for a M107 artillery shell, or for a pipe joint. The factory floor contains over 100 machines, including the foundry, metal cutters, molding presses, cooling racks, turning lathes, ring applicators, sanders and painters. Several different machines may be capable of carrying out a given operation, which raises the question: for a given product design, which is the optimum path through the factory floor, from steel billet at the entry, to shell casing at the exit ?

Traditionally, plant engineers devote days and weeks to answering this question, laying out (by hand) dozens of candidate process plans, and examining exhaustively their characteristics, incluiding plan feasibility, time spent in process, processing costs, amount of material wasted at various points, and compliance with government-mandated tolerances on various dimensions of the final product. A tool for partially automating this engineering analysis is needed, hence the term Computer-Assisted Process Planning (CAPP).

Beginning two years ago, General Dynamics decided to solve this problem by participating in the University of Florida's Integrated Product and Process Design (IPPD) program [1]. In this paper, we describe the project and the system designed to date. We also describe the IPPD program and the organization of student teams that are carrying out this project.

2. PROJECT OVERVIEW (YEAR 1).

In the first year, a proof-of-concept system was designed, centered around a two-pass algorithm to generate process plans. Data structures for representing the various machines on the factory floor were designed and implemented. The IPPD Teams were able to demonstrate that a process plan could indeed be generated given the feature information of a part. Model feature information was read in from PRO/Engineer [2] using a gateway. This information was passed to the Process Plan Generator which, coupled with a user interface, produced a number of process plans for the M107 ammunition round.

Even though the first year's efforts to create a Semi-Generative CAPP Toolset of this magnitude were quite successful, there was much room for improvement. In particular, the proof-of-concept software lacked a overall well-considered design. Last year, the focus of the project was on optimizing both the software architecture and the processes that are selected for a given part. The primary design goal for the software design has been to make it a Service-Oriented Architecture. The primary design goal for designing the process plan selection mechanism is to optimize process plans using information such as tolerance stack-ups, cost, and machining time.

The following customer needs have been established by General Dynamics:

- Implement 3D computer aided design (CAD) methodology for integrating product and process.
- Data is used to generate bills of materials (BOM's), visual work instructions, and other manufacturing documents.
- Perform simulation of manufacturing.
- Generate controller code.

The overall product design specifications include:

- Modules responsible for creating, editing, and storing process plans.
- Modules for creating and maintaining the Machine Library.
- Usage of CAD software to represent the part for path feasibility analysis.
- Usage of CAD software to generate in-process diagrams.
- Version control of all process plans and the machine library to ensure consistency.
- Fault-tolerance: must be able to handle services going down and coming back up.

The customer needs are being met by the implementation and integration of the various modules or services for which each implementation group is responsible. The modules include Machine Services, the Process Plan Generator, System Services, Data Management Services, CAD Services, and Simulation Services.

The most important service is the Process Plan Generator. Its function is illustrated in Figure 1. For each operation, several machines on the factory floor must be considered. In the process plan illustrated in Figure 1, the operations for Cut, Turn, Finish and Paint are considered, and the specific process plan involves Saw2, Lathe 1, Chemical Bath, and Automated Spray.

Operation	Candidate Processes
Cut	Saw 1, Saw 2, Saw 3
Turn	Lathe 1, Lathe 2, Lathe 3
Finish	Lathe 2, Chemical Bath, ECM
Paint	Hand Brush, Automated Spray

Figure 1. Selection of a Process Plan.

The engineer may load a pre-existing plan or create a new one, specify resources needed for the plan, specify parts, generate all feasible process plans, analyze them, validate them, and save a plan for future work.

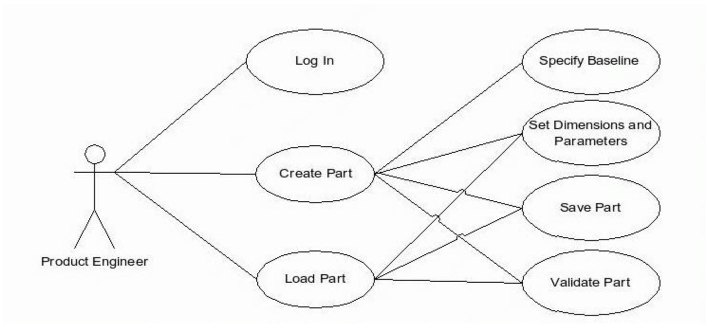

Fig 2. Usage scenario for a product engineer.

This scenario is illustrated in Figure 2, in which a product engineer may log in to work on the parts database, creating parts or loading pre-existing ones, setting dimensions and parameters for the part, and saving a part. Separately, a machine engineer can log in to work on the machine library, creating a new machine or loading a pre-existing one, establishing machine logic (e.g. reduce part diameter by 3mm.), specifying inputs and outputs to the machine, and saving the new/modified machine in the machine library. This is illustrated in Figure 3.

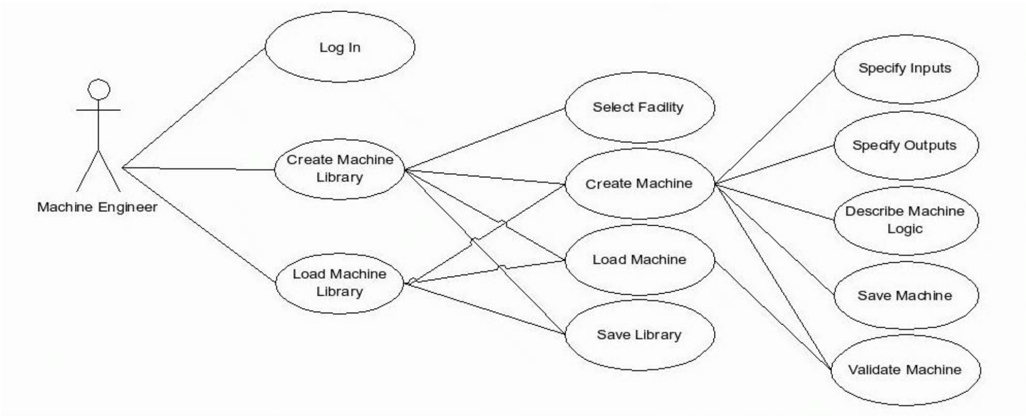

Fig 3. Usage scenario for a machine engineer.

Of special interest is the algorithm used to generate process plans. It is illustrated in Figure 4.

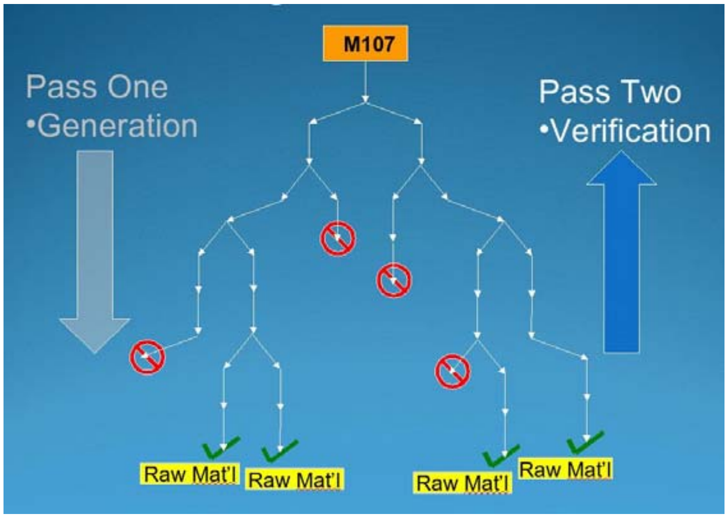

Fig. 4. Process Plan Generation and Verification

The Process Plan Generator acts in two phases or passes. In Pass 1, we begin with a target state of the manufactured part. The Machine Services component finds all machines that can produce that state. The top level of a tree is thus created, with each branch representing one machine producing the final state. The Process Plan Generator extracts the inputs of those machines, producing a new target state, and the process is repeated, with machine Services finding all machines that can produce the next-to-last state. This process is repeated until the target state is merely a steel billet, i.e. raw material, the starting point in the process. The "state" in question consists of Boolean-valued attributes, such "hot" or "painted" or "nose-cone cut". Some of these paths leading down from the root of the tree (final product) towards the raw material, cannot reach the initial state; these paths are cut from the tree as they are discovered. The end result of the first pass is a tree of possible process plans, partially pruned to eliminate "impossible" plans.

In the second pass, dimension information is added to the analysis. In the second pass, each path in the tree is examined in turn. Beginning with the initial state, Machine Services retrieves all the machines in that plan, and uses MatLab to calculate the in-process dimensions of the part. Plans are deemed to be valid unless one of the machines in it is unable to handle the dimensions of the in-process part. Process plans eliminated in this fashion are also pruned from the tree. In short, process plans are verified for "viability", i.e. the ability of all the machines in the plan to handle the dimensions of the in-coming in-process part.

Four groups of students, each with a faculty coach and a liaison engineer at General Dynamics, were organized to analyze the problems and design the system. In the next section we describe the system's architecture, the services themselves, and the breakdown of design and implementation tasks to the student groups.

3. SYSTEM DESIGN (YEAR 2).

Last year, the system was designed using a Service Oriented Architecture model. It was implemented as a collection of services deployed on application servers. This will allow it to be accessed over a network and to support multi-user access. It utilizes an HTTP server to provide web pages for the User Interface. Specifically, it will use the JBoss Application Server [3] and the Apache Tomcat HTTP Server [4]. The system uses an Oracle [5] database to store all system information including user information, parts, process plans, the machine library, etc.

Platform dependencies can often require the costly refactoring of large projects, particularly as they age. The scope of this project brings these concerns to the forefront, since much information may well be distributed at different sites and among different computers. To address this issue, a well-designed API allows support for the program to be extended to new CAD software, versioning software, simulation software, and databases. Different operating system platforms are unlikely to cause problems, since the software has been developed with JBoss, which is supported under Linux [6], Solaris [7], and MS Windows [8]. One possible disadvantage of this approach is that the CAPP system will be tightly coupled to the JBoss application server. A future decision to change application servers, (e.g. to use WebSphere), would require major re-configuration work. Nevertheless, the CAPP software relies upon (mostly) official specifications. Moving to any application server that implements these specifications, such as the Java Messaging Service [9] and Java Persistence [10], should not be too difficult. The overall system architecture appears in Figure 5.

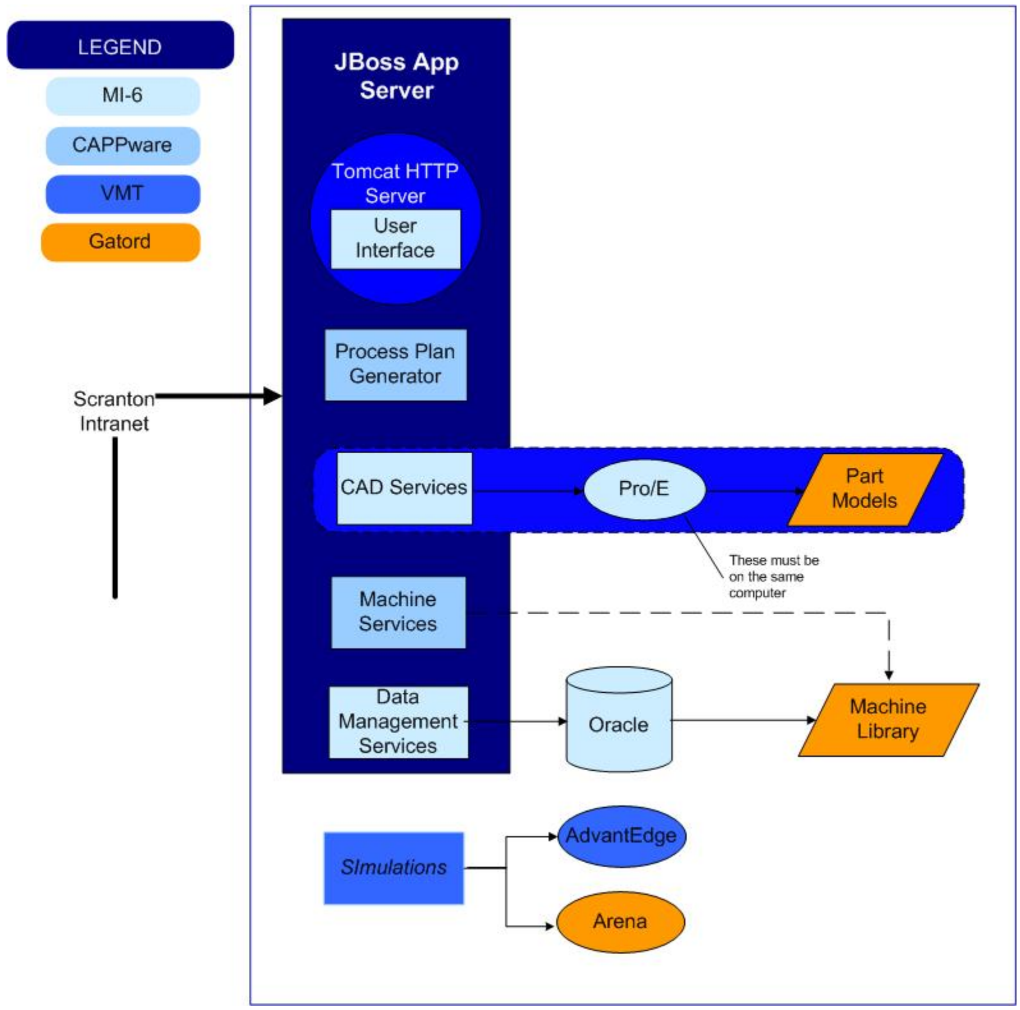

Fig. 5. CAPP System Software Architecture.

The components of the system were color-coded to illustrate the design and implementation tasks of the four teams of students. These groups were self-named as follows:

- **MI-6** : Design CAD Services, Data Management Services, and the User Interface. The User Interface is responsible for handling all interaction between users and the rest of the system. The service controls and manages the pages being displayed to the user. The service translates user input into commands to other services, and returns any necessary data to the user. CAD Services provide an interface to the CAD software. It translates CAD data into a format usable by the rest of the system. It translates calls from the system into CAD software specific commands. It also allows the system to modify CAD parts and retrieve data about the parts, including diagrams. CAD Services also provides a generic interface for CAD software, but has modules responsible for communication with specific CAD software. Data Management Services was responsible for managing the storage of information in the database. It exercises version control over various pieces of information, such as information to keep different parts of the system from becoming out-dated. It also exercises version control over the Machine Library to ensure that process plans that were generated using an out-of-date library or machines will be notified of changes, and that they should be re-run.

- **CAPPware**: Design Machine Services, the Process Plan Generator, and overall System Services. This group was responsible for the creation, editing, and analysis of process plans. It used machine and part information to generate all process plans for a given part using a specified machine library. It also allowed for analysis of process plans for metrics at the discretion of the customer. It receives machine information from Machine Services to determine path feasibility. It communicates with CAD Services to obtain in-process information about a part, to determine the criteria for machine selection. It also uses CAD Services to generate in-process diagrams. It communicates with Data Management Services to store and retrieve process plans.
- **VMT**: Design the Process Plan Simulation Services. Simulation Services is be responsible for creating 2-D and 3-D visual demonstrations of specific operations on a particular part by a specific machine. It will (eventually) be responsible for using physical machine data to generate information about the machine's manufacturing capabilities. It uses CAD software and Production software to simulate a machine.
- **Gatord**: Construction of the part models the Machine Library. The Machine Library is responsible for containing all information about the machines in the factory. There are three kinds of information: *Physical, Manufacturing, and Metric*. *Physical Information* describes physical characteristics of the machine, e.g. bed length, max RPM, horsepower, etc. *Manufacturing Information* describes information about the machines manufacturing capabilities. This includes size of acceptable part, features it can accept and create, tolerance, etc. *Metric Information* describes information about the manufacturing process. This includes time to machine a part, throughput, etc. *Physical Information* can be used to determine *Manufacturing Information. Manufacturing Information* can be used to generate process plans. *Metric Information* can be used to analyze process plans.

4. ADDITION OF PART FEATURES (CURRENT YEAR).

At the end of last year's project, we determined that we could improve significantly on the design of the system, primarily by adding the concept of a Part Feature (to include, eventually, the part's in-process state), and the concept of a Feature Library. This was motivated by the need to reseprent manufactured parts so that we account for both geometry and state. A simplified system architecture was developed, shown in Figure 6. A Product Service Interface has been developed this year, to perform the following actions:

- **Add Features or Components to a Part**
- **Get/Remove Features from a Part.**
- **Get/Modify parameters of a Feature.**
- **Add or remove state information from a Feature.**
- **Get the current state of a Feature.**
- **Generate a CAD model of a Part described by certain Features.**

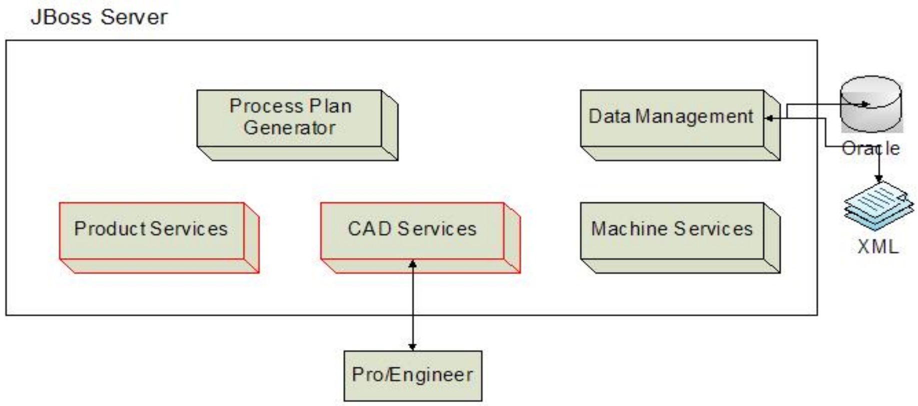

Figure 6. Simplified System Architecture.

A sample Part Feature Concept, and its corresponding Part Concept Feature Graph, are illustrated in Figure 7. The Body Feature (B) is establshed first, with datums (reference planes) D2 and D3. Then, the nose-cone Feature is added (T), with its location specified by its own datum (D2). Finally, the tail-section Feature is added (N), with its own datum (D3).

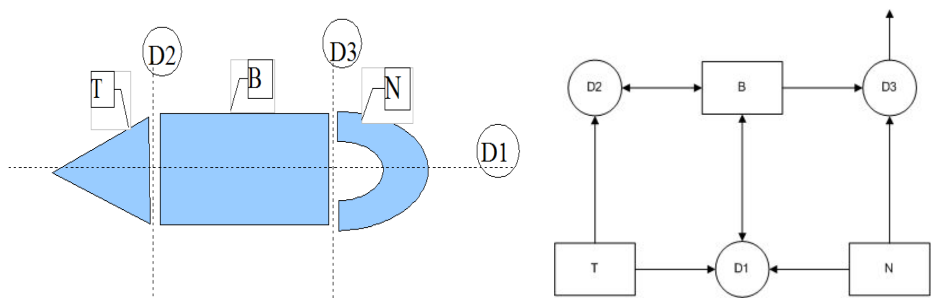

Figure 7. Part Concept for a shell casing, and its corresponding feature Graph.

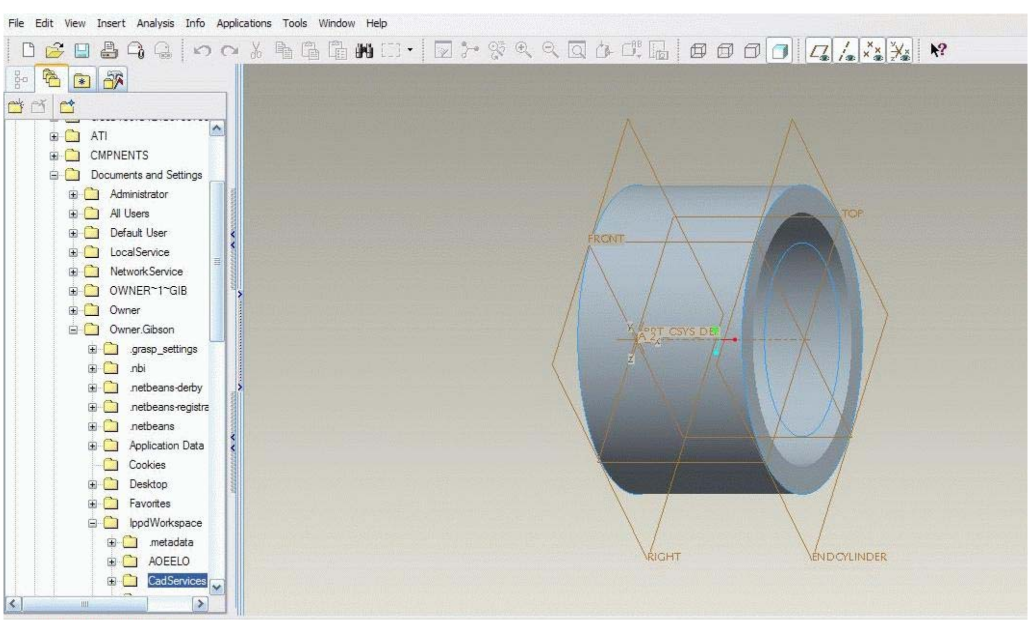

Figure 8. First step in modeling a three-Feature part: a simple cylinder.

After implementation, using Pro/E as our underlying CAD system, we managed to create a modest Feature Library. The Feature Library includes enough Features to create a realistic scenario of the modeling of a manufactured pipe-joint, with its three features. These are illustrated in Figures 8, 9, and 10.

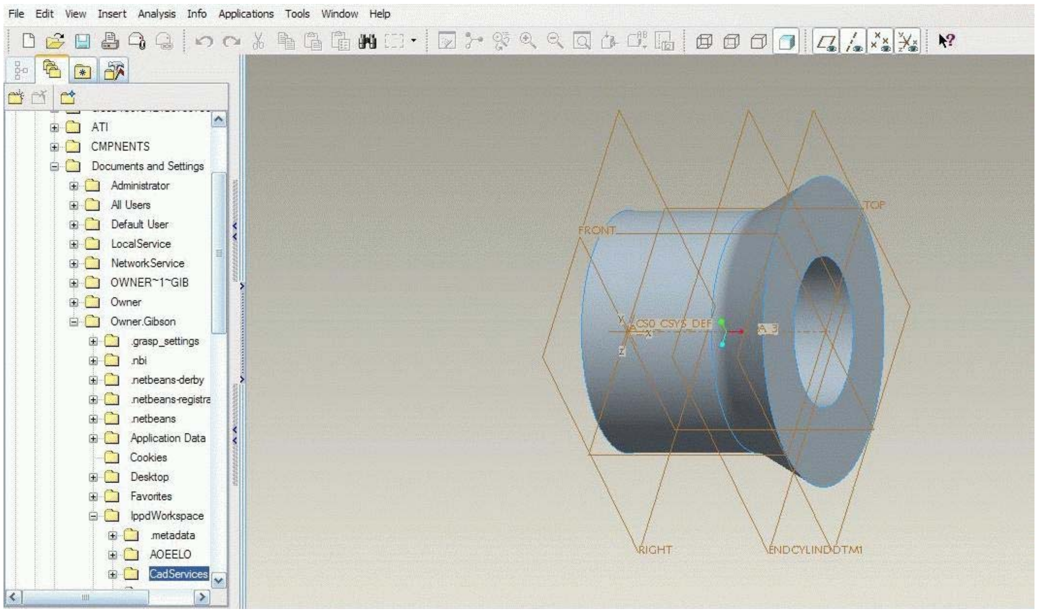

Figure 9: Second step in modeling a three-Feature Part: Addition of Neck Feature.

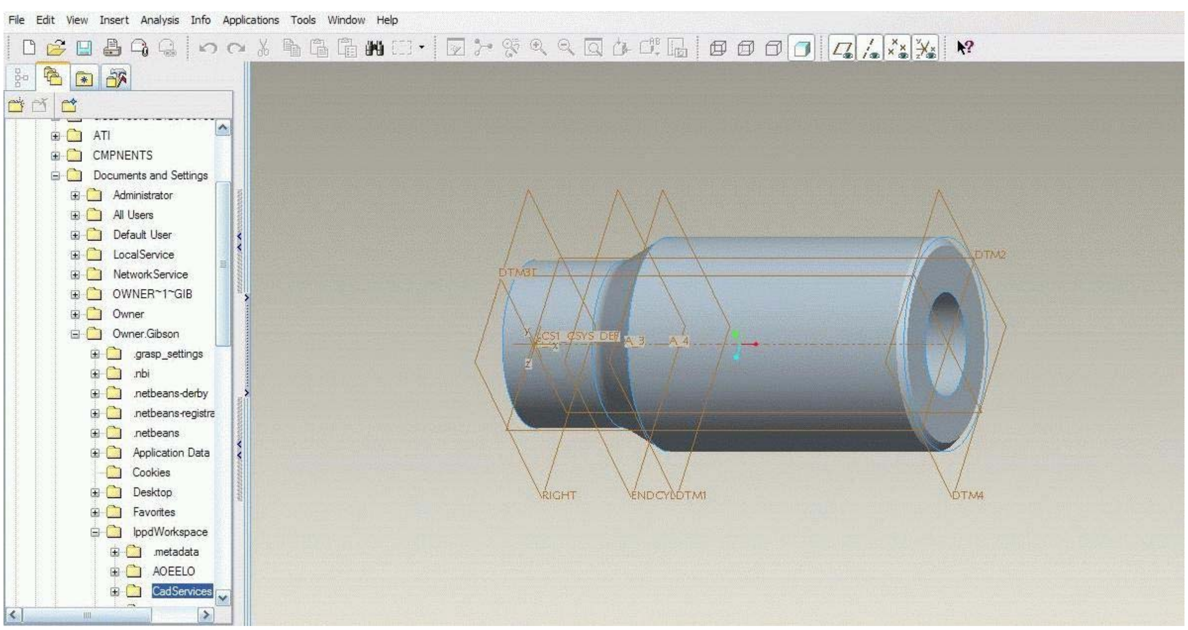

Figure 10: Third step in modeling a three-Feature Part: addition of the body cylinder.

5. STUDENT PARTICIPATION.

Each year in this project, we have had four groups of students, each with a faculty coach and a liaison engineer at General Dynamics, were organized to analyze the problems and design the system. These students enrolled in the University of Florida College of Engineering's Integrated Product and Process Design (IPPD) program. In this program, an interdisciplinary group of senior-level students, of various engineering disciplines, engage in a product or process design project for an industrial sponsor, under the supervision of a faculty coach and liaison engineers from the sponsoring company. The students spend 2 semesters in this capstone design experience, designing and building authentic products for those industry sponsors. Participating students are seniors from the Business School, and all Engineering disciplines. The Projects, technical advice, and financial support are provided by the corporate sponsors. Teams and individuals are evaluated against defined project deliverables and lecture/workshop performance. Since 1994, over 1400 students from 12 disciplines have participated in 242 projects from 62 sponsors, which include companies such as NASA, Boeing, Dell Computer, Dow Chemical, DuPont, Energizer, Florida Power Corp., General Dynamics, Harris, Honeywell, IBM, Kimberley-Clark, Kraft Foods, Lockheed Martin, Motorola, Pratt & Whitney, Raytheon, Siemens, Southern Nuclear, Sunbeam, Texas Instruments, Tropicana, and the US Air Force.

6. CONCLUSIONS.

We have presented our on-going project at the University of Florida, sponsored by General Dynamics, Inc. The goal of this multi-year project is to partially automate the creation of process plans for manufacturing. We described the nature of this multi-year project, and the current status of the design that has been produced. The design is based on the Service-Oriented Architecture, and we described this year's primary activity, the addition of a Part Feature Library. It has been an extremely rich experience for all involved, particularly for the students, who experienced working on large-scale software in a real-life situation, in an authentic industrial setting.

REFERENCES

[1] IPPD program: http://www.ippd.ufl.edu/

[2] Pro/Engineer: http://www.ptc.com/products/proengineer/

[3] JBOSS: http://www.jboss.com/products/platforms/application

[4] Apache Tomcat: http://tomcat.apache.org/

[5] Oracle: http://www.oracle.com/index.html

[6] Linux: http://www.linux.org/

[7] Solaris: http://www.sun.com/software/solaris/index.jsp

[8] Windows: http://www.microsoft.com/windows/

[9] Java Messaging Service: http://java.sun.com/products/jms/

[10] Java Persistence: http://java.sun.com/developer/technicalArticles/J2EE/jpa/