INTEGRATED PRODUCT AND PROCESS DATA FOR B2B COLLABORATION

Boonserm Kulvatunyou
Richard A. Wysk
Department of Industrial and Manufacturing Engineering,
Pennsylvania State University
310 Leonhard Building
University Park, PA, U.S.A.

Nenad Ivezic
Albert Jones
Manufacturing Engineering Laboratory
National Institute of Standard and Technology
100 Bureau Drive
Gaithersburg, MD 20899-3460, U.S.A.

Please send any comments regarding this paper to:
Boonserm Kulvatunyou, PhD.
Manufacturing System Integration Division
National Institute of Standards and Technology
100 Bureau Dr. MS 8260
Gaithersburg, MD 20899-8260
Tel: 301-975-6775 Fax: 301-975-4482
Email: serm@nist.gov

Number of Page: 37
Number of Figure: 13
Number of Table: 3

INTEGRATED PRODUCT AND PROCESS DATA FOR B2B COLLABORATION

ABSTRACT

Collaborative development of engineered products in a business-to-business (B2B) environment will require more than just the selection of components from an on-line catalogue. It will involve the electronic exchange of product, process, and production engineering information during both design and manufacturing. While the state-of-the-practice does include a variety of ways to exchange product data electronically, it does not extend to the exchange of manufacturing process data. The reason is simple; process data is usually tied to specific manufacturing resources. These resources are not known typically at product development time. This paper proposes an approach, called an Integrated Product and Process Data (IPPD), where manufacturing process data is considered during product development. This approach replaces traditional process plans, which are resource specific, with a resource-independent process representation. Such a representation will allow a much wider collaboration among business partners and provide the necessary base for collaborative product development.

Keywords: Integrated Product and Process Data, Process Planning Decomposition, Process Planning Evolution, Collaborative Product Development, Collaborative Manufacturing, Virtual Enterprise

1   INTRODUCTION

In a B2B collaborative-product-development framework, the service provider should be able to evaluate manufac-turability and cost from the product data provided by the client.   Moreover, in such a framework, the client and the service provider should be able to communicate the evaluation criteria, discuss the manufacturing content, and re-solve any issues regarding conditions (e.g., time) of service delivery and general B2B relationships.

Traditionally, collaborative-product-development approaches have relied exclusively on the product design data. However, design data does not provide sufficient information for B2B collaborative product development. There are two main reasons. First, design data does not provide the vocabulary, such as manufacturability and cost, to enable business-level communications. Second, geometric data alone is not sufficient for a number of down-stream activities such as purchasing, production planning, and scheduling, which are needed to compute delivery dates -- another crucial ingredient of B2B communications.

The objective of this paper is to propose a new approach whereby data describing a high-level work content is generated and associated with the features in the product design. This data, which constitutes a new type of process plan, supports a precise evaluation of manufacturability and cost and serves as a foundation for B2B communica-tions and negotiations. Traditional process plans are resource-specific -- the data is generated based upon available resources, associated process knowledge, and capability. The proposed approach provides a resource-independent view of the process plan called Resource Independent Operation Summary (RIOS). This view is important since the client cannot know or assume the process capability of the manufacturer. For example, a machine at one shop will have different process accuracy from the same machine at another shop due to differences in usage and environ-ment.

RIOS is a process model view of the product data and is the essential component of our proposed architecture. It captures high-level process and manufacturing requirements. RIOS-based architecture was created based on works in the process knowledge modeling by Chang and Wysk [3]; process planning decomposition by Wysk *et. al* [28]; and process representation by Catron and Ray [2] and Wysk et al [28].

 In this paper, we make use of material-removal processes -- such as hole making, roughing, and face making -- in a flexible-manufacturing domain to illustrate our approach. For this application domain, the process knowledge includes the shape-producing capability, magnitude-of-volumetric capability, compatible materials, and production

rate. This general process knowledge plays an essential role in generation of a RIOS instance from the design data. The RIOS specification schema itself consists of a number of information schemas including operation-level schema, process-level schema, graph-representation schema, and request-for-quote schema. Hence, generation of a RIOS instance implies an instantiation of these schemas in the context of a specific manufacturing task and general process knowledge.

As a conceptual example, Figure 1 illustrates design of a part to be made from alloy steel wrought with 250 Brinell hardness. Using the general process knowledge, a process engineer can establish that the material removal process should be used to manufacture the part. Subsequently, a RIOS specification can be generated for the design as illustrated in Figure 2. The figure illustrated that RIOS specification captures high-level manufacturing requirements into two levels of directed graphs.

<Figure 1 goes here>

<Figure 2 goes here>

The rest of this paper is organized as follows. Section 2 provides literature reviews of works related to the product data for manufacturing system integration and the process representation. Section 3 describes a collaborative manufacturing architecture that motivate the use of RIOS functionality. Section 4 outlines the key concepts that guided construction of RIOS. Section 5 gives important detail of the schema definitions that constitute RIOS specification. Then, Section 6 illustrates construction of a RIOS instance using an example part. Finally, Section 7 concludes the paper by summarizing the proposed architecture and the RIOS approach.

2   LITERATURE REVIEW

Traditionally, collaborative-product-development approaches have relied exclusively on the product design data. However, design data does not provide sufficient information for B2B collaborative product development. There are two main reasons. First, design data does not provide the vocabulary, such as manufacturability and cost, to enable business-level communications. These quantities must be derived from the design data. However, the technologies that can automatically map design data to manufacturing work content, which is necessary to evaluate the manufacturability and compute the cost, have been used successfully on simple products only. They fail to meet accuracy and completeness requirements needed for products that exhibit properties such as feature interaction, feature

intersection, and feature isomorphism. Furthermore, their computational complexity is still very high due to the combinatorial explosion that occurs in performing machining-volume search [26].

Second, geometric data alone is not sufficient for a number of downstream activities such as purchasing, production planning, and scheduling, which are needed to compute delivery dates -- another crucial ingredient of B2B communications. Other product representations are being defined on top of the geometric data. Among them are STEP (Standard for Exchange of Product Model Data, ISO 10303) Geometry and Topology Models Part 42 [10], Mechanical Product Definition for Process Planning Using Machining Features Part 224 [13], Feature-based Design Description Language (FDDL)--a non-standard effort by Gu and Elmaraghy [9]. These approaches will allow new features to be defined by combining a set of core features or by relating them directly to the geometric data. Although the set of core features provides better semantics for engineering, they are too detailed for B2B business-level communications. The ability to generate new, aggregate features will resolve this problem, but it may cause interpretation problems, which may lead to errors in the generation of work content. In addition, none of these approaches allows alternative representations, which would provide the flexibility of alternative semantics for the same features, thereby reducing risk of the interpretation problems.

Reviews of a number of process representations indicate that none of them satisfies the representational and functional requirements needed by RIOS as shown in the first column of Table 1 in Section 5.5. The reviews can be summarized and discussed by categorizing the process representation into five categories.

1) Process representation for system analysis. These process representations are, for example, Behavior Diagrams [1], Functional Flow Block Diagrams [4, 8], and IDEF0 [14]. The existing representations in this category have no semantic formalism; thus, they are not machine understandable (although some of them possess mathematical formalism such as Generalized Activity Network [6] and Hierarchical Task Network [7]).

2) Process representation for process documentation. IDEF3 [15] is an example of process representation in this category. IDEF3 provides a formal construct to document how to do things as well as what the system can do. Similarly, the process representation in this category merely provides a formal construct for human communication.

3) Process representation for activity planning and scheduling. The representations in this category typically have limited capability, which aims at representing the temporal relationship between tasks for planning

and scheduling of resources. Examples of the process representations in this category are the Gantt Chart and PERT Chart [21]. These representations provide no construct that would relate the task to the product information.

4) Process representation for information interchanging. The process representations classified into this category are STEP AP213[1] [12], STEP Part 49[2] [11], Process Specification Language (PSL) [16], and Process Interchange Format (PIF) [19]. PSL, STEP Part 49, and STEP AP213 (which in turn relies on the Part 49), i.e., they satisfy most of the requirements for the B2B collaborative product development shown in Table 1. However, these process representations only provide abstract level construct to identify the resource and the process capability requirements without specifying what those capabilities are.

5) Process representation for manufacturing control. Examples of process representations in this category are the AND/OR directed graph by Wysk, Peters, and Smith [28] and ALPS (A Language for Process Specification) by Catron and Ray [1].

Wysk, Peters, and Smith specified a formal process plan schema associated with the manufacturing resource information that is necessary to control a computer integrated manufacturing system. This process plan schema provides sufficient information to evaluate the manufacturability, manufacturing cost, and product completion time. Although the schema has syntactic formalism based on the AND/OR directed graph, it lacks of semantic formalism. The purpose of the schema is for shop floor control; thus, it is has no task composition/decomposition feature. ALPS was expected to be an integration protocol among process, production, and execution in a fully automated CIM system. It defines the information schema for specifying the information, resources, products, and materials that are required for performing manufacturing processes in terms of object entities (i.e., in an object-oriented manner).

APLS is extended from the AND/OR Directed Graph representation and the process-view of GASP simulation language. Particularly, the hierarchical abstraction is a notable solution to the conflicting goals of simplicity, explicitness, and completeness of the process specification that are important to RIOS. For example, an intelligent system would accept high-level specification and search for more information such as resource requirements from the given plan, eliminating the need for explicit resource allocation information. Although ALPS provides a rich construct, its

---

[1] Process planning representation for numerical control machining
[2] Process planning representation, structure and properties

emphasis is on the temporal relationship between tasks and events. ALPS also lacks of the linkages to resource capability requirement, process capability requirement, and product data.

Although, the process representations identified are powerful for its application areas, they are lacking of one or more representational capabilities necessary for the B2B collaborative product development. The rich syntax capability is typically important for low-level process representation where events and interactions between resources need to be handled. RIOS schemas developed in this research addresses all the process representation requirements identified in of Table 1.

3 RIOS-BASED MANUFACTURING COLLABORATION ARCHITECTURE

Our main argument is that a work-content definition is necessary for B2B collaborative product development. Current technologies and past research efforts do provide with accurate and repeatable approaches to work-content generation. A key element of the proposed collaboration architecture is that a work content can be specified and encoded independent of the target manufacturer, then sent to any manufacturing partner for automatic evaluation and negotiation. Such a work content specification, which we call RIOS, must be resource independent and integrate product data with process data.

Figure 3 illustrates an example, collaborative, B2B Automatic Vendor Selection (AVS) architecture using RIOS. In this figure, a product designer, the client, has designed a part and needs to select a supplier to produce that part. The client generates the RIOS specification of the design by using the generic process knowledge of the application domain. A request-for-quote -- consisting of production requirement, design data, and RIOS data -- is then distributed to different manufacturing (vendor) agents to evaluate time, cost, and manufacturability.

<Figure 3 **Goes Here**>

Each manufacturing agent must be able to understand RIOS so that its Resource Specific Process Planning (RSPP) module can generate a Resource Specific Routing Summary (RSRS) using input from the Manufacturing Resource (MR) model and the Process Knowledge (PN) model (see Figure 3). The MR model is an information model that characterizes the available equipment -- machine tools, robots, humans, cutting tools, fixtures -- that the vendor could select to manufacture the finished product. The MR model must represent the technological and economical specifications for that equipment. For example, the technological specifications of a machine tool may include table size, repeatability, and capacity; the economical specifications, on the

other hand, of that same machine tool may include utilization cost, number of such machines, and operating status, as described by Kulvatunyou and Wysk [18].

A PN model characterizes the process capabilities of the vendor. This model may be decomposed into shop-level knowledge and machine-level knowledge. Shop-level knowledge declares the processes the vendor can execute and the best attainable performance. This knowledge can be expressed in terms of high-level vocabularies such as shape producing capability, quality, and process accuracy. Machine-level knowledge consists of both declarative and procedural knowledge. The declarative knowledge expresses the process capability specific to each resource, possibly in combination with other resources; the procedural knowledge expresses how resources should be utilized to achieve a set of process requirements.

The RSPP module uses the MR and the PN models to determine when a given RIOS requirement matches the vendor's core competency. When a match occurs, the RSPP module generates the RSRS, a resource-specific specification of work content. For example, the RSRS of a vendor agent that results from a hole making process requirement may consist of twist drilling and finished reaming sequences while RSRS of another agent may consist of twist drilling and semi-finish boring sequences. Once the RSRS is generated, the vendor agent can respond to the client with delivery time and cost computed using, say, the activity-based cost accounting procedures. A Manufacturing Control (MC) model, which represents the production strategy and activity flow of the manufacturing system, may be used to generate the cost and time estimations by taking Master Production Schedule and system dynamics into consideration. If not, the agent may initiate a negotiation process with the client or third party vendors

4   RIOS FUNDAMENTAL CONCEPTS

In this section, we describe four concepts essential to the construction of RIOS: the process hierarchy that provides fundamental semantics for the RIOS specification, the Universal Level Process Knowledge (ULPN) that includes only common process knowledge not specific to resources, the decomposition of process planning decisions that assures resource-independence during RIOS construction, and, the augmented AND/OR directed graph (digraph) that provides a representation scheme for RIOS.

4.1 Hierarchical Modeling of Process Specification

The Webster Dictionary [20] provides seven definitions for the word "process". There are three generic definitions and three definitions directed to specific fields such as biology, law, and manufacturing. In the most appropriate

definition for this context, a process is "a series of actions or operations conducted to achieve an end". This definition implies that a process may consist of a single operation with its own result or several actions, with each of these actions consisting of one or more such operations.

Figure 4 illustrates the process hierarchy, inherited from Ray [23], conceptualized in this research. First, processes are clustered into two different types: self-driven and intentional. A self-driven process occurs by itself, for example, a natural or biological process. This type of process is not relevant to this research. An intentional process is actuated purposely by some mechanism, often human. In manufacturing, we can decompose these intentional processes into two types: business processes and fabrication processes. Business processes deal primarily with the integration and manipulation of information. Fabrication processes deal primarily with the transformation and transportation of materials. As such, they define the limitations on producibility and manufacturability of a given facility; hence, they are the focus of this research. In this paper, we limit the discussion to material transformation, specifically material removal processes. We believe that the results can be extended easily to material transportation.

<Figure 4 **Goes Here**>

The process hierarchy in Figure 4 provides a semantic basis for RIOS scalability. That is, other types of processes can be added to the schema, by subtyping the entities in the hierarchy, and definitions can be incorporated into the RIOS specification by following the procedure illustrated in Section 5.

4.2 Universal Level Process Knowledge

Process knowledge can be organized into three levels: a universal level, shop level, and machine level [3]. The Universal Level Process Knowledge (ULPN) plays a significant role in RIOS because it contains only knowledge that is common to every resource capable of implementing that process. The Manufacturing Advisory Service (MAS) created by UC Berkeley associates [27] is an excellent repository of ULPN. Furthermore, MAS has demonstrated that a small subset of appropriate manufacturing approaches can be identified from conceptual design information. Examples of conceptual design information are overall shape, type of material, type of geometry, and overall precision. The ULPN also includes knowledge to perform economical analysis of processes such as minimizing the cost of selected processes, minimizing the number of handlings, and maximizing the ease of handling. The use of ULPN will be illustrated in Section 6, where example RIOS construction is detailed. Two other levels are the shop level, which refers to the attainable process capabilities of a shop, and the machine level, which refers to the machine-specific

process capabilities. Because they are resource specific, they are not used to generate RIOS; they are, however, used to process the RIOS and obtain RSRS by the manufacturing vendors.

4.3 Process Planning Decision Decomposition

A resource-independent process plan, which can be used to evaluate product manufacturability at different facilities, must not presume resource availability and capability of those facilities. Thus, the decomposition of process planning into resource-independent decisions and resource-dependent decisions is critical. The resource-independent decisions are used to generate RIOS; the resource-dependent decisions are used to map RIOS to a resource-specific plan. Two prior efforts are relevant to our work:

- Wysk et al. [28] divided process planning into three steps: 1) feature graph (precedence graph) generation, 2) AND/OR graph of CLDATA (Cutter Location DATA), and 3) AND/OR graph of machine/tooling/fixture combinations (NC File). Although CLDATA is portable to different machining equipment, only the first step of this process planning decomposition is resource independent. Once the CLDATA is generated, the specifications of equipment and tooling have already been defined and the processes that satisfy the accuracy requirements are already selected as well.

- Shah and Mantyla [25] stated that tasks in process planning could be divided into the following subtasks: 1) Operation planning: process selection, 2) Setup planning: grouping of operation, 3) Fixture planning: selection of work holding, and 4) NC program generation: machine motion control program generation. It can be seen that the first two steps are in the reverse sequence as proposed by Wysk et al. [28]. Shah and Mantyla left the process precedence until the second step and grouped them according to the tool access direction.

Although these two decompositions provide a high-level description of a good sequence of decision-making, they lack sufficient detail for our purpose -- separation into resource-independent and resource-dependent decisions. To do this, we need to augment these decompositions with several important constraints.

1. Precedence constraints due to geometric constraints: tool access direction and work holding orientation

2. Precedence constraints due to relocating the part adversely affecting the repeatability requirement of a part, e.g., refixturing resulting in loss of positioning accuracy

3. Precedence constraints due to geometric tolerance requiring extreme repeatability, e.g., reference surfaces must be machined before the specified surface [29]

4. Precedence constraints due to economical rationalization of machining (these are typically side effect constraints), e.g., machining rough surface with rough (inexpensive) process before applying delicate (small and expensive tool) machining process; sequencing the process to reduce tool chattering

5. Process accuracy constraints, i.e., selection of required processes to achieve the specified accuracy. The result is a process precedence due to an unidirectional chain of process operations [29]; for example, when a hole needs to be bored or reamed, it must be drilled first.

6. Technological constraints: resource availability including available tools, fixtures, and machines (e.g., machine criteria may include acceptable part size, power, kinematic capability.)

7. Process economy: selection of operation based on minimum cost and/or time.

The decisions regarding the first four constraints are resolved based on part geometric data, tolerance specification, and ULPN with minimum reflection on the resource and process capability specifics. Therefore, these four constraints are resolved at the time of RIOS generation. The other three decision criteria will be resolved at the time of RIOS mapping. Although interdependencies may exist among constraints, these can be handled by the alternative plan representation in RIOS (described in Section 4.4).

4.4 RIOS Representation

Since a product can be machined in several ways, RIOS must have the flexibility to allow alternative processes to be specified whenever they can be justified by the ULPN. Therefore, an augmented AND/OR directed graph (digraph) representation is used to facilitate that flexibility. This graph representation will be detailed in Section 5.2.

The RIOS specification is a requirement specification. That is, it only indicates the process and manufacturing capabilities necessary to produce the product. A requirement specification would consist of statement like, "The hole on this product requires a hole making process that can achieve 0.5" diameter and 0.008" positioning accuracy". A descriptive specification would consist of statement like, "The hole on this product needs twistdrilling at 0.4687 inch diameter and then boring with 0.5 inch diameter".

5   RIOS SPECIFICATION

RIOS specification schemas are divided into five parts to ensure extensibility and flexibility. These schemas are: (1) the `RIOS_Support` schema, (2) the `RIOS_Transaction` schema, (3) the `RIOS_Operation_level_information` schema, (4) the `RIOS_Graph` schema, and (5) the `RIOS_Process_level_information` schema. The support schema

defines information entities in support of (i.e., shared by) other schemas. The graph representation schema provides precedence and alternative constructs to specify the requirements and processes defined in the operation information and process information schemas. The following subsections describe and partially illustrate each schema using the EXPRESS-G graphical representation [24] (complete models with EXPRESS formal syntax can be found in [17]). Appendix A illustrates the EXPRESS-G notations.

5.1 RIOS_Support Entities

Entities defined in the RIOS_Support schema are used by other schemas. These entities include measurement entities and workpiece specification entities among others. Figure 5 illustrates the schemas of `Workpiece` and `Length_measure` entities.

<Figure 5 **Goes Here**>

5.2 RIOS Graph Representation

The `RIOS_Graph` schema employs an AND/OR directed graph representation with extensions in the logical connection, task composition, task decomposition, and the process oriented-nodes to capture the process plan alternatives, hierarchical tasks, and high-level process plan information. Since the RIOS schema represents high-level information only, the graph schema does need all of the capabilities described in Wysk et al. [28]. For example, it does not need to represent time duration, resource allocation, exception handling, error recovery, and cost data.

The RIOS graph representation essentially consists of AND, OR, and GROUP logical connections. The OR junction represents alternative processes or operations. The OR junction provides flexibility to the resource requirement evaluation. The AND junction represents alternative sequencing of the processes where both branches need to be visited. It has two subtypes--`Serial_and` and `Parallel_and`. The `Parallel_and` has another degree of freedom to execute the tasks simultaneously. The AND junction typically reflects alternative processing styles. The GROUP junction indicates that the processes within the connection may be combined into a single processing sequence.

RIOS is represented at two levels: Operation Level Graph (OLG) and Process Level Graph (PLG). Each node in the OLG specifies setup, types of resources required for all processes within the operation, and tolerance equations. Processes are associated to each operation by assigning a PLG to each node of the OLG. Each node in the PLG specifies the process capability requirements (e.g., dimensions and accuracy) to shape the workpiece into a finished

product. PLG integrates process data with the product data by referencing the geometry from within the process requirement.

5.3 Transaction information

The `RIOS_Transaction` schema provides support for the request-for-quote interactions between a client (e.g., design house) and a manufacturing vendor. The application objects defined in this schema include business information and production requirements (business objectives such as order quantity and expected delivery time). The schema also defines application objects that encapsulate RIOS information necessary for manufacturability and cost evaluation such as part model and RIOS graph. The base entities in this schema are the `RFQ_Transaction` and the `Quote` entities.

5.4 Operation level information

The `RIOS_Operation_level_information` schema defines information for equipment capability requirements including kinematics, setup, work holding capacities, and tolerance equations (variables in the equations will be associated with those in the process level information). Figure 6 partially illustrates the schema. The schema has a hierarchical structure with each entity having either no label or an `ABS` label. The entities labeled with `ABS` are abstract entities; those with no label are instantiable (concrete) entities.   One proceeds down this hierarchy choosing among `ABS` entities that may exist at each level until only instantiable entities remain. Section 6 will illustrate the use of this schema.

<Figure 6 **Goes Here**>

5.5 Process level information

The abstract entities in this schema (Figure 7) associate processes to those operations modeled in the operation level schema. For example, if the material-removal operation is specified, only the material-removal process subtypes can be associated with the operation. Each process is a unit of operation where several passes may be needed to achieve the requirements in a process entity. In addition, each of these process entities is not a composition of others by definition. For example, each `Hole_making_process` is a single hole. A composite hole should be specified by placing several `Hole_making_process` within a `GROUP` connection. Then a graph-based pattern matching can be used to recognize the process composition.

<Figure 7 **Goes Here**>

Figure 8 illustrates the schema of `Roughing_process` entity as an example of several processes shown in Figure 7. The definition of each process essentially consists of geometric, topological, tolerance, material, and initial process information. Geometric information is defined by referencing the geometry in the product data. The roughing process is defined by a set of adjacent removal volumes. Only the definition of extrusion volume is shown in the figure. It should be noted that the set of adjacent volumes does not indicate the removal volume decomposition (with respect to a specific process), but it merely indicates the volumes that need to be removed and that may be combined.

<Figure 8 **Goes Here>**

Table 1 summarizes that RIOS satisfies an extensive collection of key requirements for process representation.

<Table 1**Goes Here>**

6   EXAMPLE RIOS CONSTRUCTION

This section illustrates the use of ULPN to generate RIOS graph for an example part given in Figure 9. Several pieces of information, which significantly affect the cost of making a product, are available at the early design stage. Shown in the list below, they can be used to sketch an operation-level graph (next to each criterion is the value associated with the example product). Some of the parameters below are results of another criteria, such as the small batch size regulates setup cost and setup time to be low in order to maintain an acceptable cost.

- **Batch size**: 10

- **Overall shape and geometry**: Prismatic

- **Bounding box volume**: 7.5in^3

- **Material**: Hot-rolled carbon steel wrought medium carbon, 350 bhn;

- **Minimum dimensional tolerance**: 0.001in

- **Minimum surface roughness**: 125µin

- **Minimum wall thickness**: 0.25in

- **Per part cost**: Low

- **Setup cost**: Low

- **Setup time**: Low

<Figure 9 **Goes Here>**

For each node in an operation level graph (OLG) according to the `RIOS_Operation_level_information` schema in Figure 6, the first step is to identify whether a business or a manufacturing operation is required. If the manufacturing operation is required, then the type, material transformation or material transportation, must be selected. Since the material transportation is not in the scope, the OLG consists of solely the material transformation operations.

Proceeding to the next step, the type of material transformation must be selected; currently, only material removal and geometry changes are considered. The following subsections discuss these two types of operations with respect to the parameters listed above.

6.1 The Geometry Change Operation

Casting, extrusion, forging, sheetmetal forming, and injection molding are among the common types of geometry-change operations used in production. Although sheetmetal forming and extrusion operations may meet most geometry and accuracy requirements, sheetmetal forming requires ductile material and extrusion has a 2.5D geometry limitation. The example part is a true 3D part, requiring machining on top and bottom, and it is made from carbon steel, which does not have ductile properties.   More importantly, most geometry-change operations are meant for a very large batch production and high production rate due to the very high setup time and cost associated with the die and/or pattern. This is a major parameter determining that the geometry change operation is not a favorable choice for the example design and production requirements.

6.2 Material Removal Operation

There are four types of operations in the 'material removal operation' category. They are chemical removal, thermal removal, electrical removal, and mechanical removal. The kinematic capabilities of these operations make them applicable to a wide range of geometries, scales, materials, and precisions. The setup cost associated with these operations is low because setup equipment can be configured differently and reused. Setup time can vary considerably depending on the tolerance requirements and the geometric complexity of the workpiece. The example part will take only minutes to set up using a low cost and simple rectangular vise. One characteristic of the material removal operation that differentiates it from other types of material transformations is the relatively slow production rate. These characteristics make the material-removal operation suitable for the small to medium batch size production (1-500

pieces). After considering each of the four material-removal-operation subtypes, we have selected the mechanical-removal operation for the example part. We justify this selection in the rest of this section.

Examples of chemical-removal operations are chemical etching and electrochemical machining. These operations are clearly **not** suitable for the example part, because they typically operate on silicon composite materials in the micro- to nano-meter range of the overall part dimension.

Examples of common tools for thermal-removal operations are torch (or gas), plasma, and laser. This type of operation typically produces a very coarse surface finish, and it can only be used reliably to cut depths of 0.5 inch or less [5]. In addition, the underlying heat permeation makes it difficult to control final dimensions. Consequently, it is used only to remove stock material quickly. The only exception is laser cutting, which, while expensive, has several advantages over mechanical cutting including material independence, high-quality surface finish, oblique removal direction, and dimensional accuracy. Nevertheless, these capabilities are unnecessarily costly for the example part. We conclude that thermal removal is **not** a viable option.

Examples of electrical-removal operations are Electric Discharge Machining (EDM) and sputtering. These operations slowly remove a thin layer of material by using either electric spark to boil the material or electron particle to collide with the material. This type of operation, which typically removes material in the micro- to nano- level regime, may be used to cut special features into parts of varying sizes, materials, and geometries [22]. Although this type of operation can easily obtain nanometer precision, the underlying heat limits the surface quality to the same level as that of mechanical-removal operation. In addition, setup time is high and setup cost is medium. Consequently, we conclude that these operations are not needed for the example part.

A mechanical-removal operation imparts a shear force through sharp cutting edges to remove material. This type of operation is versatile because of the sophisticated motion control to approach the workpiece from multiple directions and the variety of tool geometries that can remove material from the workpiece. Setup time, which depends on the workpiece geometry and the required rigidity. Setup cost is usually low, since the same set of tools can be reconfigured differently. The removal rate, which depends on the required tolerances, is considerably quicker, if not the quickest of all material-removal operations. Because of the moderate production rate, setup time, and setup cost, the processes in this category are generally suitable for a small to medium batch size from single to hundreds of pieces of production. The bounding box volume can range from 0.1 inch^3 to 3000 inch^3. Complex contours, free form surfaces, and gear shapes can be produced.   Tolerances of +/-0.005 inch can be reached easily, but +/-

0.001inch requires more control. Tolerances in the four-digit range are achievable, but it is significantly more costly [27]. Since 16µin *surface finish* is achievable, the required 125µin surface finish on the example part is considered easy to achieve. Thin-wall surfaces, however, are a limitation; 0.1in thickness over a minimum of 5in span can be machined. Once again, we have selected the mechanical-removal operation for the example part.

6.3 Constructing the Operation Level Graph

The example part consists of a `slot` feature, two `steps`, one `counterbored` hole (consisting of a `core` hole and a `c-bore` hole), and a `4-hole pattern` (see Figure 10). The part needs at least two operations (setups) because the slot is on the opposite side from the other features -- we will refer to the side on which the slot is as the 'bottom' and the opposite side as the 'top'. The first constraint in Section 4.3 deals with work-holding and tool-access directions; therefore, the first question is which side and which features to do first. Usually, answers to these kinds of questions are based on the economical rationalization, which are affected by the number of material-handling setups, machining operations, process accuracy, and total cost. In the example part, processing the bottom side first simplifies the material handling operations. On the other hand, processing the top side first allows the step to be processed before the hole pattern. This simplifies the cutting process, because there will be no interruption during the step machining. In addition, the tool chatter reduces, because the depth of the hole decreases. Therefore, regardless of the resource-specific data, the OLG will be constructed such that the operations on the topside precede those on the bottom side.

<Figure 10 **Goes Here**>

During the first setup, the `hole pattern`, both `steps`, and the `counterbored` hole will be machined. The question is, which order. We already established that the steps would be done before the holes. Furthermore, the entire hole pattern must be processed in the same setup because relocating the part will adversely affect the positioning repeatability (the 2nd constraint). To determine the remaining order, we must consider both datums and stack-up. The step (the boss's side face), which produces datum `D`, is positioned with respect to datum `C`. The `core` hole is positioned with respect to datum `D`, and the `c-bore` hole is then positioned with respect to the `core` hole. Therefore, the tolerance stack-up (Figure 10) from the `step` processing to the `core` hole and the `c-bore` hole must be taken into account. The result of the tolerance stacking causes the process accuracy requirement of the `steps` (`M20`) to be one order of magnitude higher (0.0025 inch) than the regulated constraint (`C20 = 0.02`). Thus, an alternative path (an 'OR' connection) should be given to process the `steps` and the `hole` pattern in a separate operation from the `counter-`

`bored` hole. This alternative allows 0.02" tolerance (`M20`) to process the steps and the 0.0025" tolerance allowance (if `M32` and `M42` are allocated equally) to process the `counterbored` hole -- this is the same as process in a single setup case. We note that the tolerance equations are associated with the `Tolerance_equation` entity as indicated in operation schema in Figure 6. Figure 11 summarizes the operation level graph. Table 2 summarizes the data associated with operation 3. The `surface id` interlinks this process data with the design data.

<Figure 11 **Goes Here**>
<Table 2 **Goes Here**>

6.4 Constructing the Process Level Graph

There must be four PLGs associated with each of the four operations in the OLG; however, only the PLG associated with operation 3 (`OP3`) will be illustrated here. The `OP3` consists of only the `counterbored` hole with a different datum system from operation 1 and 2. Figure 12 illustrates the process level graph (`PLG3`) for this operation. In this graph, two `Hole_making_process` nodes are grouped using the `GROUP` connection node representing the `counterbored` hole. The `GROUP` connection indicates that it is possible to combine the two processes into a single sequence by using a composite tool-geometry. The `GROUP` connection also implies that the removal volumes of the process within the connection are adjacent. Table 3 provides a summary of the data associated with the core hole from PLG3. Notice that the variable `M32` is associated with the true position in the `Positioning_with_tolerance` row. The value is assigned after the tolerance equations in the operation level graph are resolved by the RSPP module.

<Figure 12**Goes Here**>
<Table 3 **Goes Here**>

6.5 Usage within a Collaborative Framework

Once completed, the RIOS data package can be shipped with the product data package to the manufacturing vendors. It can be used for collaborative activities such as request-for-quote or automatic-vendor-selection, as illustrated in Section 3. It should be noted that one important benefit of using RIOS to communication engineering requirements is it provides a language for negotiation. The manufacturability result can be directed toward specific process and even specific process requirement, which cause trouble and money. The client side (who requests for quote) can use the manufacturability and cost estimation result (which is also detailed for cost associated with each process and operation) to negotiate the cost he/she believes should be lower (an expected cost threshold might be set upfront). While the original manufacturability result might be generated by minimizing the lead-time, the manufac-

turer side, when asked to lower the cost, may reevaluate the manufacturability by minimizing the cost. In other events, the manufacturer may try to change planning criteria and subcontract some operations to other manufacturers, whose capabilities match the requirements better and can produce a lower cost.

7   CONCLUSION

In this paper, we described the use of integrated product and process data (IPPD) for B2B collaborative product development. We argued that, because traditional process data are resource specific, a set of schemas and methodologies for representing resource-independent processes was needed.   We called this data RIOS. RIOS not only facilitates the evaluation of manufacturability, but also provides a set of vocabularies for business and manufacturing negotiation. We used an example part to illustrate the RIOS data that could be generated during the product development process, the proposed schemas, the process planning task decomposition, and the universal-level process knowledge. Although the example part was for metal-removal machines, we believe that the developed schemas may be extended to other manufacturing processes. Another part of work, which details the resource specific process planning from the RIOS data, is its final development stage, and it is making a promising result.

APPENDIX A: EXPRESS-G SYMBOLOGY

<**Figure 13 Goes Here**>

REFERENCES

1. Ballard, C.W. (1991), *Basics of Behavior Diagrams*, Ascent Logic Corporation, San Jose, CA, 1989.

2. Cantron, B. A., and Ray, S.R., ALPS: A language for process specification, *International Journal for Computer Integrated manufacturing*, 4 (2), 105-113.

3. Chang, T.C. and Wysk, R. A. (1985), *An Introduction to automated process planning systems*, Prentice Hall.

4. Defense Systems Management College (DSMC) (December 1986), *Systems Engineering Management Guide*, December 1986.

5. Duane Jaworski & Associates Web Site. Available online via <http://www.djmetalrep.com> [accessed January 2001].

6. Elmaghraby, S.E. (1977), *Activity Networks: Project Planning and Control by Network Models*, John Wiley & Sons, New York.

7. Erol, K., Hendler, J., Nau, D.S. (March 1994), *Semantics for Hierarchical Task Network Planning*, Technical Report, Computer Science Dept., ISR, UMIACS, University of Maryland, College Park.

8. Grady, J.O. (1993), *System Requirements Analysis*, New York: McGraw Hill.

9. Gu, P. H., and Elmaraghy, H. A. (September 1989), A Feature-based Design Description Language, *ASME Design Engineering Division* (Publication), 17, 53-63.

10. ISO 10303 Part 42 (1994), *Industrial automation systems and integration -- Product data representation and exchange -- Part 42: Integrated generic resources: Geometric and topological representation,* International Organization for Standardization.

11. ISO 10303 Part 49 (1995), *Product data representation and exchange: Part 49: Application Protocol: Integrated generic resources: Process structure and properties*, International Organization for Standardization.

12. ISO 10303 Part 213 (1995), *Product data representation and exchange: Part 49: Application Protocol: Integrated generic resources: Process structure and properties*, International Organization for Standardization.

13. ISO 10303 Part 224 (2001), *Industrial automation systems and integration -- Product data representation and exchange -- Part 224: Application protocol: Mechanical product definition for process planning using machining features*, International Organization for Standardization.

14. Knowledge Based System, Inc., *The overview of IDEF0: Function modeling method*. Available online via < http://www.idef.com/overviews/idef0.htm> [accessed January 2001].

15. Knowledge Based System, Inc., *The IDEF3 Process Flow and Object State Description Capture Method Overview Process Description Capture Method*. Available online via <http://www.idef.com/overviews/idef3.htm> [accessed January 2001].

16. Knutilla, A., Schlenoff, C., Ray, S.R., Polyak, S.T., Tate, A., Cheah, S.C., and Anderson, R. C. (May 1998), Process specification language: An analysis of existing representations, *National Institute of Standards and Technology. (NISTIR Document No. 6160)*.

17. Kulvatunyou, B. (2001), *A Resource Independent Process Representation for Enterprise-based Engineering Integration*, Ph.D. Thesis, Pennsylvania State University.

18. Kulvatunyou, B., and Wysk, R.A. (July 2000), "A Functional Approach to Enterprise-based Engineering Integration", *Journal of Manufacturing System*, 19 (3), 156-171.

19. Lee, J., Gruninger, M., Jin, Y., Malone, T., Tate, A., and Yost, G. (1996), The PIF Process Interchange Format and Framework Version 1.1, *Technical Report Working Paper No. 194*, MIT Center for Coordination Science.

20. Merriam Webster Dictionary (2001). Available online via <`http://www.m-w.com` (2001)> [accessed January 2001].

21. MI Standards Committee, and Duncan, W.R., Director of Standards, *A Guide to the Project Management Body of Knowledge,* Project Management Institute, Upper Darby, PA, 1996. Available online via <http://www.pmi.org/pmi.publictn/pmboktoc1.htm>.

22. NMRC Non-traditional Manufacturing Research Center (2000). University of Nebraska-Lincoln. Equipment and facilities for NMRC. Available online via <`http://www.unl.edu/nmrc/equ.html`> [accessed January 2001].

23. Ray, S.R. (1987), Process Reasoning, *Computers in Industry*, Elsevier Science Publishers B.V. (North Holland), 9, 329-335.

24. Schenck, D., and Wilson, P. (1994), *Information Modeling the EXPRESS Way*, Oxford University Press, New York.

25. Shah, J.J., and Mantyla, M. (1995), *Parametric and Feature-Based CAD/CAM*, John Wiley & Sons, New York.

26. Subrahmanyam, S., and Wozny, M. (1995), Overview of Automatic Feature Recognition Techniques for Computer-aided Process Planning, *Computers in Industry*, 26 (1), 1-21.

27. Wright, P., Manufacturing Advisory Service. Integrated Manufacturing Lab. University of California, Berkeley. Available online via `<http://cybercut.berkeley.edu/mas2/index.html>` [accessed August 2000].

28. Wysk, R.A., Peters, B.A., and Smith, J.S. (1995), A formal process planning schema for shop floor control, *Engineering Design and Automation*, 1 (1), 3-19.

29. Wysk, R.A., *An Automated Process Planning and Selection Program: APPAS*, Ph.D. Thesis, Purdue University, 1977.

INTEGRATED PRODUCT AND PROCESS DATA FOR B2B COLLABORATION

BOONSERM KULVATUNYOU is currently a guest researcher at the National Institute of Standards and Technology (NIST) from Oak Ridge National Laboratory. He recently received his Ph.D. from the Pennsylvania State University, University Park, in 2001. He received his MS from Columbia University, NY, in 1997, and his BS from Chulalongkorn University, Bangkok, Thailand in 1995. He is a senior member of the Society of Manufacturing Engineers (SME) and Computer and Automated Systems Association (CASA) of SME. He is also a member of the American Association for Artificial Intelligence (AAAI). His research interests include computer-integrated manufacturing system, simulation of manufacturing system, enterprise and e-business integration, and information modeling. His email address is <kbserm@psualum.com>.

NENAD IVEZIC is a guest researcher at the National Institute of Standards and Technology from Oak Ridge National Laboratory (ORNL). He received his M.S. and Ph.D. degrees from Carnegie Mellon University and B.S. degree from University in Belgrade. His interests include Artificial Intelligence, Distributed Systems, Supply Chain Management, and Enterprise Systems Integration. He currently works on developing a Manufacturing eBusiness Testbed, participates actively in standards development for e-business technical specifications, and explores advanced approaches for enterprise systems integration. At ORNL, he was a Principal Investigator for agent-based systems development for Lockheed Martin and the Defense Logistics Agency and for the Collaborative Management Environment for Department of Energy.

RICHARD A. WYSK currently holds the Leonhard Chair in Engineering at the Pennsylvania State University. Prior to his current position, he was director of the Institute for Manufacturing Systems and holder of the Royce Wisenbaker Chair in Innovation at Texas A&M. he has also served on the faculty of Virginia Tech and worked in industry as a research analyst for Caterpillar Tractor Co. and as production control manager for General Electric. He received his Ph.D. in industrial engineering from Purdue University in 1977 and his BS and MS in industrial engineering and operations research from the University of Massachusetts (Amherst) in 1972 and 1973, respectively. He is a decorated Vietnam veteran and author of several textbooks. Honors recognizing his research include the Institute of Industrial Engineers' David F. Baker Distinguished Research Award and the Society of Manufacturing Engineers' Outstanding Young Manufacturing Engineering Award. His research interests include computer-integrated manufacturing, computer-automated manufacturing, computer-aided process planning, and concurrent engineering.

ALBERT JONES Dr. Jones manages the Manufacturing Enterprise Integration Program at the National Institute of Standards and Technology (NIST). Prior to that, he led several projects to investigate the functional and integration requirements for the next-generation simulation tools. Before this, he worked with Dr. Rabelo from Ohio University and Dr. Yih from Purdue University to develop an Artificial Intelligence based real-time and reactive scheduling system. Prior to this assignment, Dr. Jones spent several years as Deputy Director of the Automated Manufacturing Research Facility at NIST. His earlier research includes the development of system architectures for shop floor control, cell control, and distributed scheduling. Dr. Jones has published numerous articles on control systems, scheduling, and simulation. Before coming to NIST, Dr. Jones taught for several years at John Hopkins University and Loyola College of Baltimore. He received his MS in Mathematics and PhD in Industrial Engineering from Purdue University. Dr. Jones is currently on the Executive Boards for the Winter Simulation Conference and the Engineering School at Loyola of Baltimore. He has Chaired or Co-chaired several international conferences, and has served on several proposal evaluation panels for National Science Foundation, NIST, and Defense Adavnce Research Agnecy.
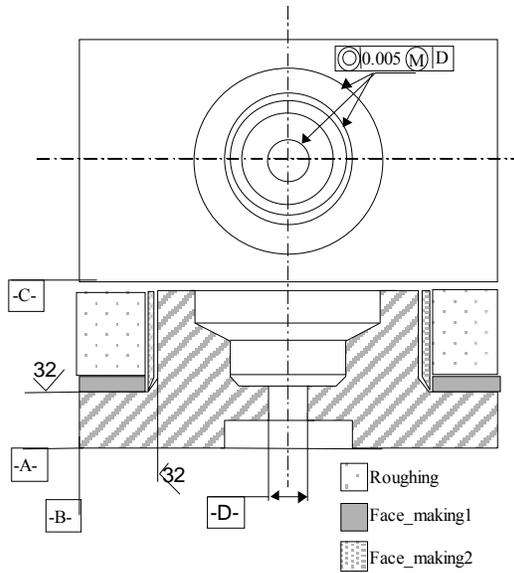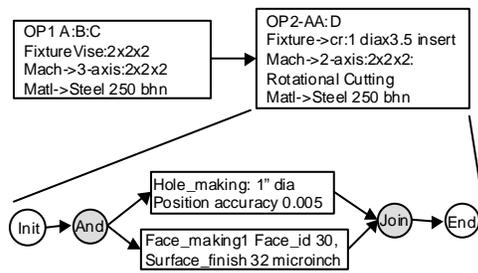
Figure 1: An Example Part Design

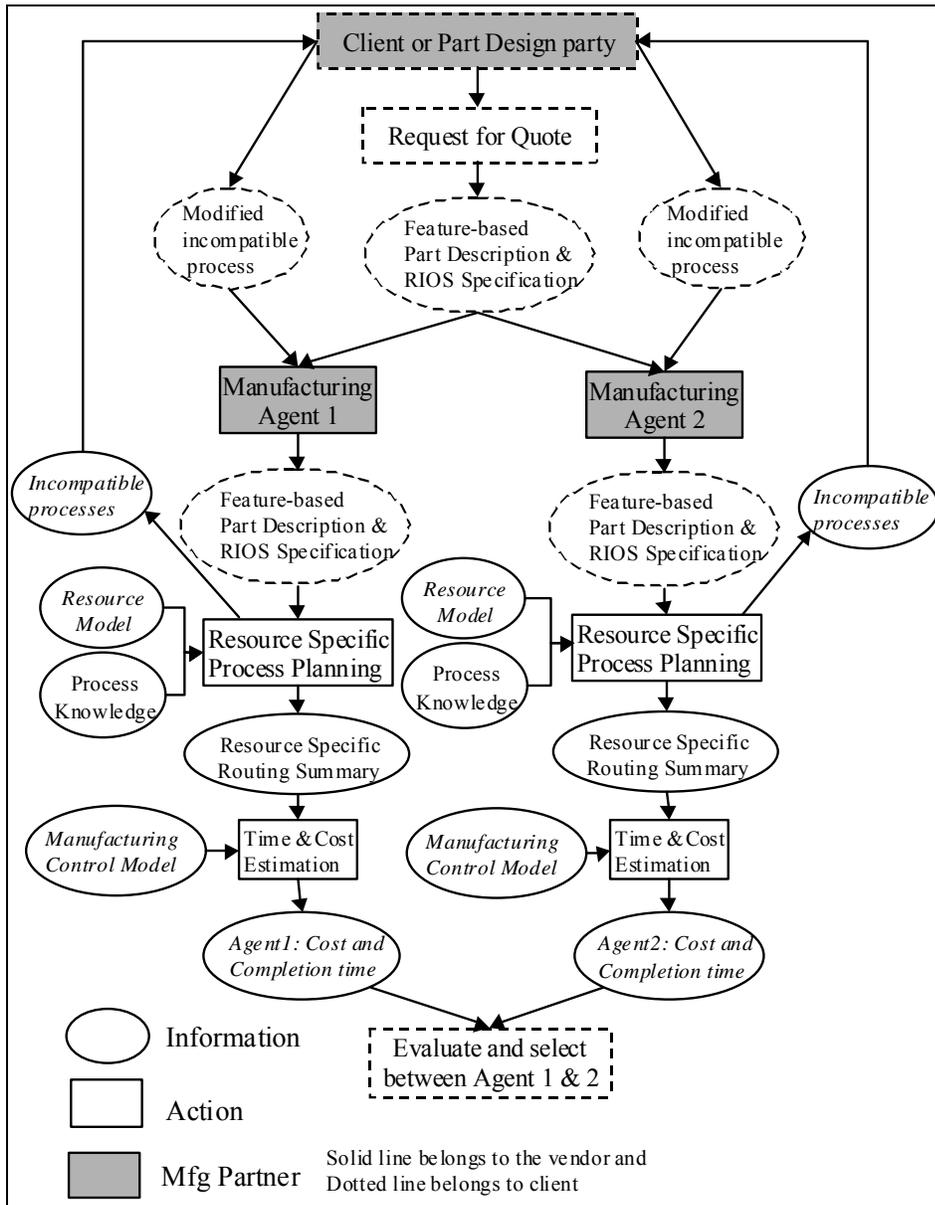**Figure 2: An Example RIOS Requirement Specification**

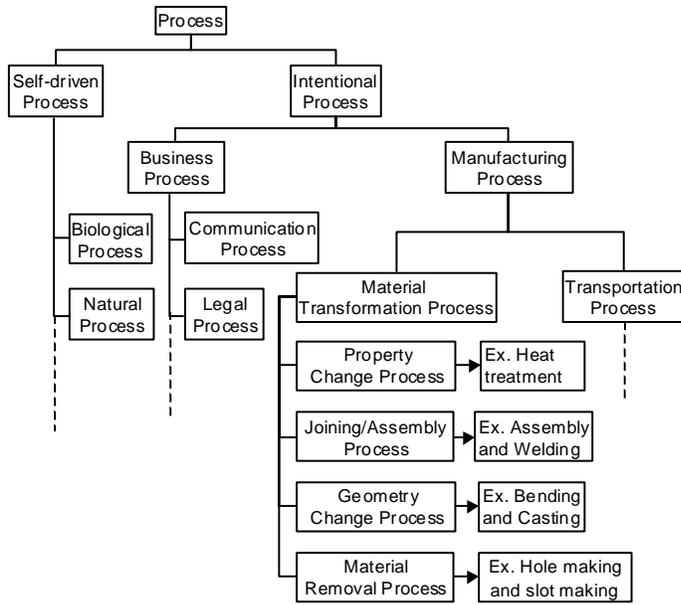**Figure 3: Web-based Automatic Vendor Selection Architecture**
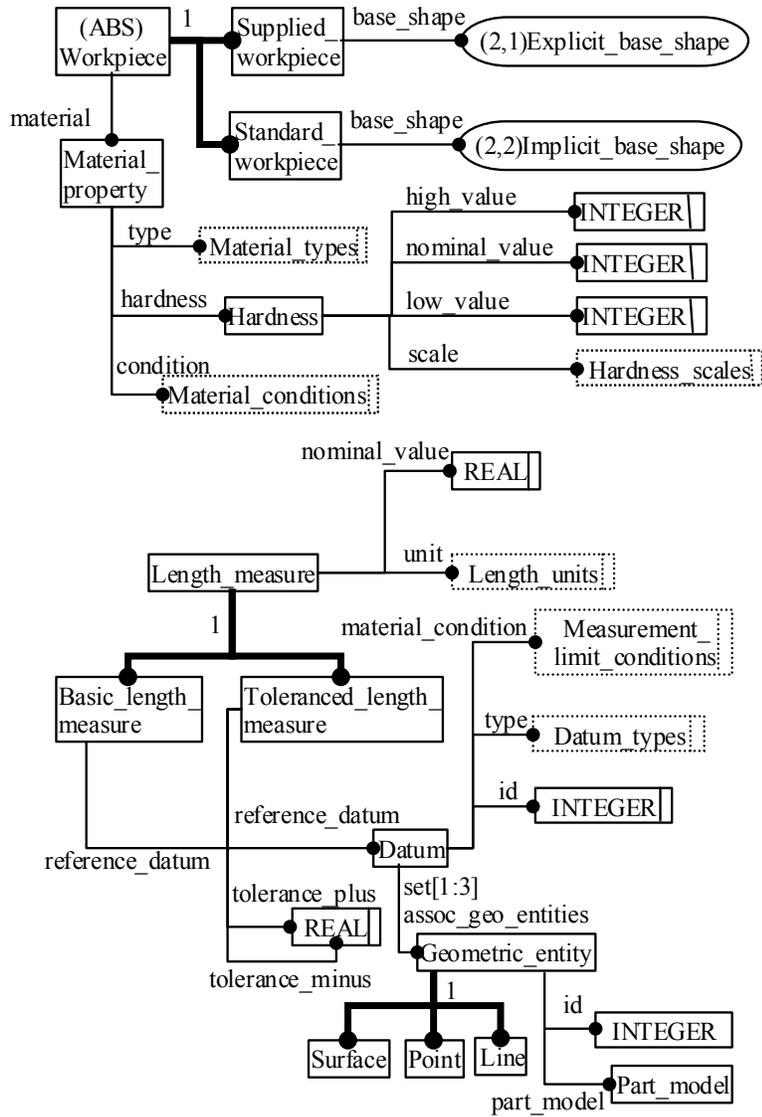
Figure 4: Process Hierarchy
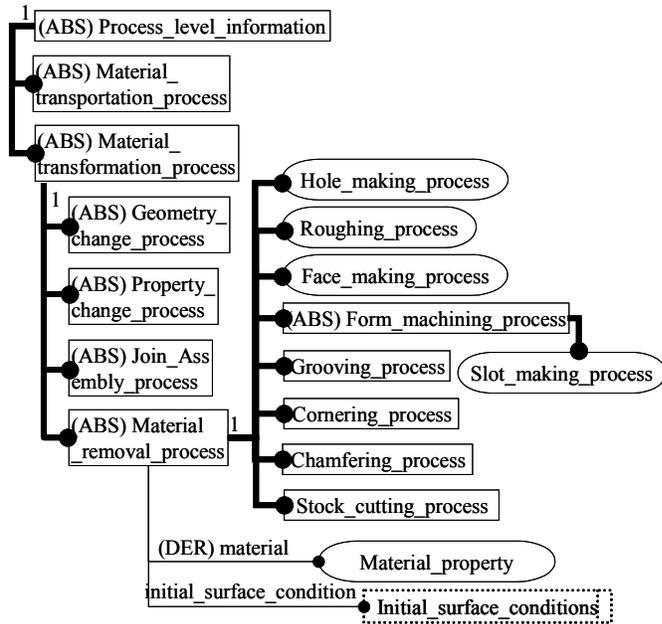
Figure 5: Example Entities in RIOS_Support Schema.

Figure 6: Example entities in RIOS_Operation_level_information schema.

**Figure 7: Example Process Entities in the RIOS_-
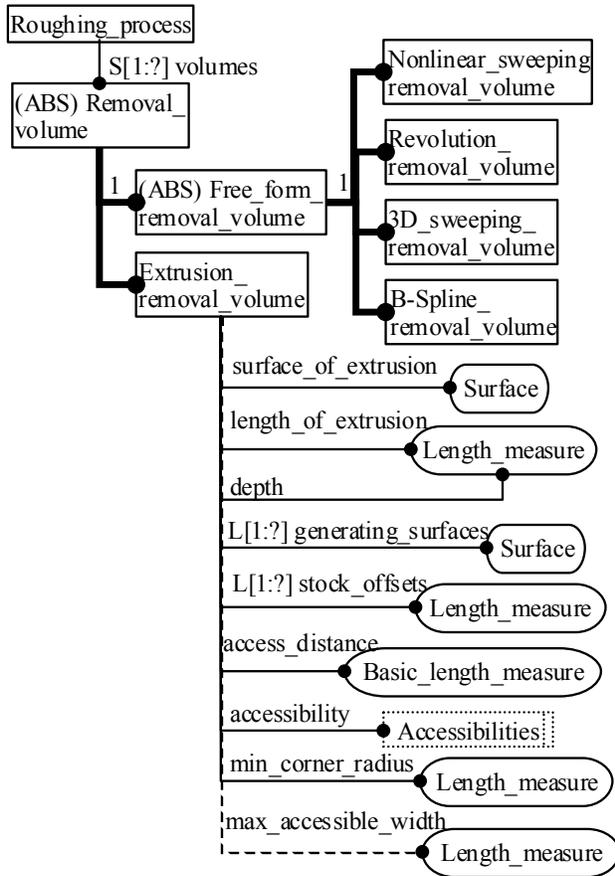Process_level_information schema**

Figure 8: Schema of Roughing_process Entity

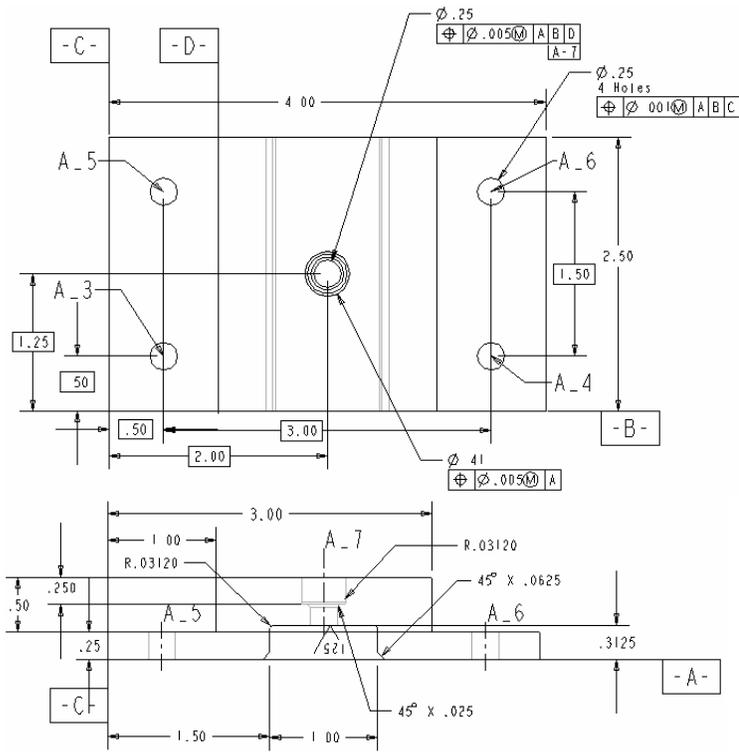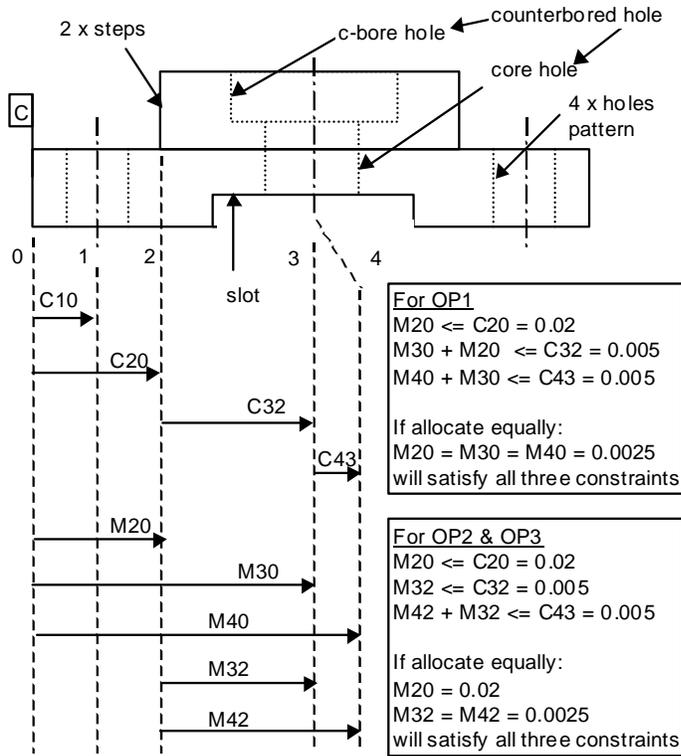**Figure 9: Example Part Design, a Laser Positioning Bracket.**
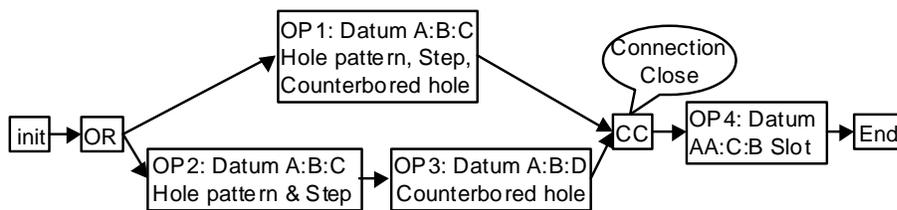
Figure 10: Tolerance Chart of the Example Part.

For OP1
M20 <= C20 = 0.02
M30 + M20 <= C32 = 0.005
M40 + M30 <= C43 = 0.005

If allocate equally:
M20 = M30 = M40 = 0.0025
will satisfy all three constraints

For OP2 & OP3
M20 <= C20 = 0.02
M32 <= C32 = 0.005
M42 + M32 <= C43 = 0.005

If allocate equally:
M20 = 0.02
M32 = M42 = 0.0025
will satisfy all three constraints

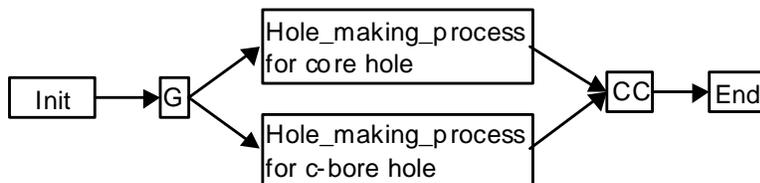**Figure 11: OLG of the Example Part.**
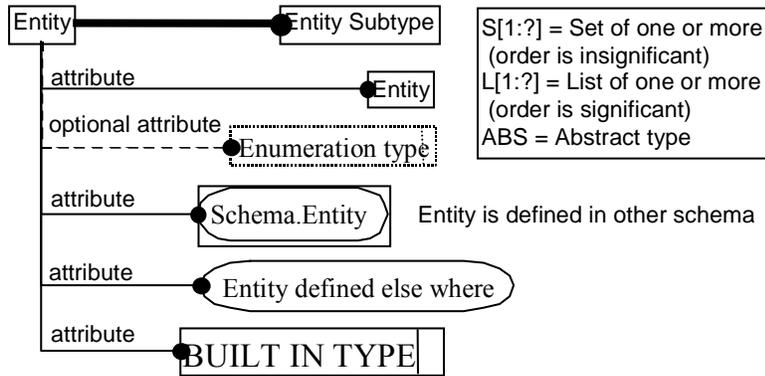

Figure 12: PLG3 of OP3

Figure 13: Express-G Symbols

Table 1: Summary of RIOS Characteristics against the Representational and Functional Requirements of Process Representation.

Requirement	Influence	RIOS compliance
Representing resource capability requirement [16]	Since the process representation is used across the enterprise, using resource identification to identify equipment is not applicable. It is necessary that a process representation allows resource capability requirement to be specified so that specific resource can be selected at a specific facility.	The RIOS_Operation_level_information schema.
Representing process capability requirement [16]	Similar to the resource capability requirement, each member in the enterprise possesses different process capabilities (even if they have the same set of equipment). The specification of process capability requirements allows members of the enterprise to select processes based on their own process knowledge, capability, and practice.	The RIOS_Process_level_information schema.
Representing alternative task [16]	A number of process planning decisions are made in order to generate a multi-enterprise process plan. Although the RIOS fundamental concepts identified in Section 4 of this paper gear toward a resource independent process plan, it may happen that a clear selection cannot be made regardless of resource specific knowledge. The alternative task representation is necessary to embrace this issue.	The RIOS_Graph schema.
Representing task sequential relationship [16]	Several process constraints resulting from design specification necessitate sequential relationships between processes.	The RIOS_Graph schema.
Representing Task composition and decomposition [16]	The characteristic of resource independent process representation will be a high-level representation. The task composition and decomposition indicate the conflicting goal of simplicity, explicitness, and completeness of the process specification [2] to be resolved with facility specific information.	The structuring of a process plan graph into the operation level graph and process level graph represents the task decomposition. In addition, each process entity defined in the process level information schema can be decomposed into several processing sequences before achieving the process requirements. The GROUP connection entities in the RIOS_Graph schema support the task composition.
Machine understandability and Internet compatibility	The machine understandability refers to the semantic formalism that enables finite interpretation of the data. The Internet compatibility refers to ability to communicate through the Internet. These two requirements are necessary features for the enterprise-based engineering integration, since it is geographically separated and it may involve business level negotiation.	The RIOS specification was first formally encoded in the EXPRESS information modeling language, which captures the semantics of the terms. The specification is then encoded in the XML framework for Internet compatibility, although XML DTD does not capture several semantic aspects of the specification. A better information schema , such as XML Schema, is evolving in the World Wide Web community to enable semantic level communication.
Integrating a resource independent process model view with the product data	The design data alone is not sufficient for automated process planning due to its underlying ambiguity for manufacturing interpretation. A process model view of the product provides additional information for automated process planning. The process data should be interlinking with the product data so that the product information is modular and facilitates information sharing.	RIOS characterizes the specification by the methodologies, described in Section 4, that gear the specification toward resource independence. RIOS also associates with the product data by referencing the entities in the product data rather than defining new geometric entity from design data.
Providing vocabularies for business to manufacturing collaboration	This functionality is necessary for enterprise-based engineering integration in the business and manufacturing level.	The RIOS schema provides not only abstract level of process descriptions, but also low level descriptions that maps directly to the manufacturing cost.
Extensibility to cover larger application domain [16]	Since manufacturing is a large domain, a semantic basis and a mechanism that allows the extensions based on the semantic basis will facilitate enterprise level collaboration. The semantic-based methodology is becoming a prevalent technology in the knowledge-sharing domain.	The RIOS specification is developed with an object-oriented modeling mechanism. Comprehensive abstract level taxonomies allow RIOS to be extended while maintaining the semantic relationship with new entities.

Table 2: Summary of Data Associated OP3 Node

Data name	Value
operation_name	Mill_slot
workpiece_holding_size	Block_bounding (WxLxH) = 2.5" x 4.0" x 0.74"
datum_system	A, Plane, surface id = 10
	B, Plane, surface id = 22
	D, Plane, surface id = 41
fixture_requirement	Fixture_type = Standard_fixture
	Difficulty_level = Simple
min_number_of_translational_kinematics	2
min_number_of_rotational_kinematics	0
tolerance_constraints	M32 <= 0.005", M42 + M32 <= 0.005"

Table 3: Summary of Data Associated with PLG3

Data name	Value
Process_level_graph	id = Alcoa_bracket1_PLG3, init_node_id = node3.1
Init_node	node_id = node3.1
Group_node	node_id = node3.2, prev_node_id = node3.1
Manufacturing_process_node	node_id = node3.3, prev_node_id = node3.2, process_name = core_hole
Manufacturing_process_node	node_id = node3.4, prev_node_id = node3.2, process_name = cbore_hole
Connection_close_node	node_id = node3.5, prev_node_id = node3.3, node3.4
End_node	node_id = 3.6, prev_node_id = node3.5
Hole_making_process:	process_name = core_hole
Processing_material	derived from RIOS_Transaction work-piece's material property
Initial_surface_condition	Smooth
Initial_surface_orientation	Flat
Feature_direction	Forward
Bottom_condition	Through_hole_bottom_condition without transition
Diameter	0.25" +/- 0.01, RFS Datum axis A_8 with associated Line id = 311 is the Datum
Depth_of_size	0.75"
Positioning_with_tolerance	True position with M32 tolerance at MMC
	RFS Datum plane A with associated surface id = 10 is a Primary_datum
	Secondary_position = 1.25" basic length measure located from a Secondary datum, which is a RFS Datum plane B with associated Surface id = 22
	Tertiary_position = 1.00" basic length measure located from a Tertiary datum, which is a RFS Datum plane D with associated Surface id = 41
Placement_surface	Surface id = 36
Access_distance	0.00" Basic_length_measure from Datum AA with associated Surface id = 36
Access_width	Diameter of the counterbored hole can interfere the tool access; thus, this field is a Length_measure having a value equal to the counterbored hole diameter = 0.4062"
SF_side	125 microinches
Resulting_cylindrical_face	Surface id = 281
Top_transition	45 degrees chamfer from Datum AA. Chamfer depth = 0.025" from the same datum