NISTIR 5910

Unified Process Specification Language: Requirements for Modeling Process

Craig Schlenoff
Amy Knutilla
Steven Ray

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards and Technology
Manufacturing Engineering Laboratory
Manufacturing Systems Integration Division
Gaithersburg, MD 20899

NIST

NISTIR 5910

Unified Process Specification Language: Requirements for Modeling Process

Craig Schlenoff
Amy Knutilla
Steven Ray

U.S. Department of Commerce
Michael Kantor, *Secretary*
Technology Administration
Mary L. Good, *Under Secretary for Technology*
National Institute of Standards and Technology
Arati Prabhakar, *Director*

September 1996

Table of Contents

Abstract

A wide range of applications deal with the manipulation and expression of collections of activities. Examples include project management, workflow management, business process reengineering, product realization process modeling, manufacturing process planning, production scheduling, simulation, and Computer Aided Software Engineering, each of which is supported by some combination of graphical programming and control languages, Petri nets, PERT charts or other representation methodology. Each of these applications serves a specific audience and need, and focuses on particular aspects of a process. Nevertheless, much could be gained by sharing information among applications. One of the primary obstacles to such integration is the lack of any common representation of what is really the common underlying concept of process. The objective of the work described here is an investigation of the feasibility of a unifying specification of process which is applicable to all of the above applications, yet powerful and robust enough to meet each set of requirements. This document represents the results of the first phase of the work – that of researching the process representational requirements for design/manufacturing process life-cycle applications. These requirements are categorized into four categories; core, outer core, extensions, and application, which aided in describing the role of the requirements in the overall challenge of process representation.

1.0 Introduction

The motivation of this work is to move closer to the goal of integrated manufacturing engineering and control systems. This goal has been elusive, despite the great strides taken in information technology in the past few decades. To achieve integration (and better yet, integratable systems) requires at least compatibility of data representations, communication paradigms, and system architectures. Advances have been made in each of these areas, such as product data exchange (STEP - ISO 10303 [36]), communication protocols (TCP/IP [3], OSI [5]) and architectures (CIM Framework [23], CORBA [64]). Although it is the information that must be shared between these systems, it is the representation and the language that provides the mechanism to allow the sharing to take place.

One area of data representation which has received relatively little attention, when compared to product data, is process data. This is particularly interesting when one considers that every aspect of a manufacturing enterprise involves some form of process. Just as significant is that each manufacturing function dealing with process typically has its own representational approach. Thus, a production scheduling system typically has its own means of entering, manipulating and representing a sequence of actions, a process planning system has another, and a workflow management system yet another. Clearly there are benefits to be gained from a representation which is common to all of these applications as shown in Figure 1.

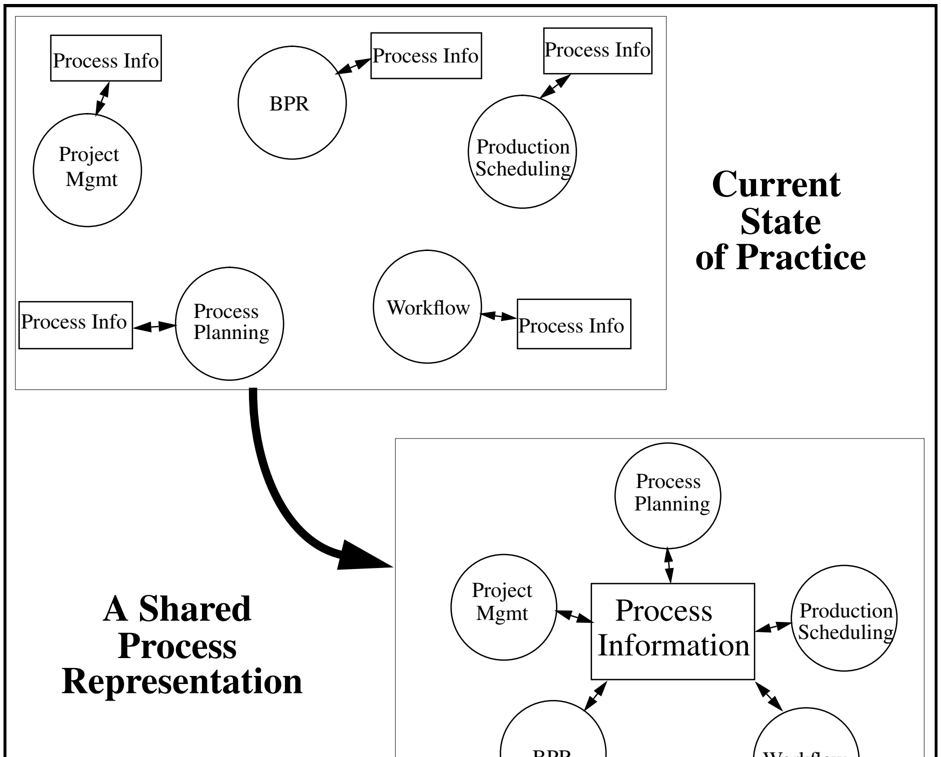

Figure 1: Motivation for Project

Motivated by this growing need to share process information in the manufacturing environment, this project will ultimately result in a generic process specification language (PSL) for describing process, building upon existing methodologies. This paper presents the requirements for specifying the business and engineering processes found in the manufacturing environment, ideally laying a strong foundation for the development of a common, unifying process specification language.

1.1 Problem Statement

In the last decade, there has been an increase in the number and types of software applications which attempt to capture the essence of process. These range from tools that simply portray processes graphically to tools that enable simulation, analysis, and/or control of processes. As manufacturing companies move toward increased integration, there is a growing need to share process information. For example, project management software will use process data from workflow applications. This is leading to the conclusion that as more processes are modeled, analyzed, monitored and controlled, and as more of these automated applications are implemented and integrated, there must be a robust, standard, formal method for representing processes.

Several recent industry roadmaps and agendas underscore this need for standard, formal process representation.

The National Research Council (NRC). At the request of the National Science Foundation, sixteen members from industry and academia, all members of NRC's Computer Science and Telecommunications Board or the Manufacturing Studies Board, prepared a report to determine the computer science and engineering research priorities to support advanced manufacturing. The resulting report [13], "Information Technology for Manufacturing: A Research Agenda", states:

> "... tools for describing process are critical for the design of individual products, the design and operation of factories, and the development of modeling and simulation technology. Formal descriptions are necessary if processes are to be represented in sufficient detail and with enough specificity to be adequately complete and unambiguous; such formalisms would allow designers to describe factory processes (involving both machines and people), design activities, decision processes, among others."

Identified research needs in this area are:

- A language for expressing process descriptions, translatable across technical domains
- Process model representation schemes
- Specific process models that reflect all relevant spatial and temporal transformations

The National Electronics Manufacturing Initiative (NEMI). The Manufacturing Information and Manufacturing Systems (MIMS) roadmap [53] states "process representation languages and terminology" as a core competency need in the area of software technology. The report states the importance of a common process representation, along with other standards for interfaces, networks, and product dictionaries, as necessary for the integration of factories.

The National Center for Manufacturing Sciences (NCMS). Recently, NCMS published a report [52] which presents the view of "leading manufacturing industry experts who are well positioned in the industry to understand the forces that will influence North American manufacturing." In the report, the authors state that the

> "...modeling techniques available today allow reasonably accurate depiction of process operations and factory-level flexibility, for example, and can improve their performance. However, one of the problems with these techniques is their lack of inter-operability..."

This need to support interoperability is underscored later within tables of identified manufacturing needs, among which were:

- Methods including common representation of enterprise
- Capture of nominal and variant process behavior
- Concurrent process planning and resource allocation
- Interoperable planning tools using speech, video, text, photo, etc.

The Semiconductor Industry Association (SIA). The National Technology Roadmap developed by SIA and distributed by SEMATECH, Inc. [62] cites several high priority technology needs which would be addressed by a common process representation, such as

- Integrated model-based factory simulation/control systems
- Reusable manufacturing requirements and design information
- Integrated models (and supporting techniques) for evaluating factory design alternatives and operational support

Semiconductor Research Corporation (SRC). The SRC Factory Sciences Technical Advisory Board [22] has identified what they termed "operational" or "manufacturing" models as critical to their manufacturing science strategy position. By these terms they meant to identify the need for modeling the behavior of factories including movement of lots, people, tools and consumables, inventory levels, etc. Applications which would be supported by such models included integrated scheduling, capacity planning, cost models, information and material flow.

Researchers and practitioners from academia, government, and industry (users and vendors) have made several attempts to formally represent process. For example, a tool called CACE/PM (Computer Aided Concurrent Engineering/Process Modeler), developed by Perceptronics[1] and based on Modified Petri Net methodology and notation, has been used to model an advanced aerospace mission. ISO 10303-213 (Draft International Standard for Numerical control process plans for machined parts), an application protocol of STEP (Standard for the Exchange of Product Model Data), is being integrated into many computer aided process planning (CAPP) software packages including MetCAPP and CS/CAPP, and is being used as a process representation by multiple private industries. An informal consortium of academic and industry researchers has developed a Process Interchange Format (PIF) to share and interlink heterogeneous business process models, and an DARPA (Defense Advanced Research Projects Agency) project at Knowledge Based Systems, Inc., is extending PIF to enable the translation of distinct process representations into a single environment for the development of new, collaborative business processes.

There have also been attempts by government agencies to integrate various applications using a common process representation. There are three government funded projects at Raytheon Electronic Systems that all use a model of manufacturing process, namely Integrated Product and Process Initiative (IPPI http://agile.cimds.ri.cme.edu/IPPI), Integrated Process Planning/Production Scheduling (IP3S http://agile.cimds.ri.cmu.edu), and Shared Integrated Product Process Development (IPPD http://webhost.sainc.com/arpa/am3/). Inherent to all of these projects is "the use of a common representation for exchanging process planning and production scheduling information."[58] IMPPACT (Integrated Modelling of Products and Processes using Advanced Computer Technologies), a project in ESPRIT (European Strategic Programme for Research and Develop-

1. No approval or endorsement of any commercial product in this paper by the National Institute of Standards and Technology is intended or implied. This paper was prepared by United States Government employees as part of their official duties and is, therefore, a work of the U.S. Government and not subject to copyright.

ment in Information Technology) attempted to develop and demonstrate a new generation of integrated modelling systems for product design and process planning." [15] While there are documented shortcomings of all of these approaches and methodologies, there is definite progress toward improved formal representations of process to address the growing need.

Finally, there are likely numerous "unanticipated" uses for a formal, common representation of process. For example, Boeing Helicopters' Advanced Computing Technologies group has identified a need for a common process representation in order to better apply their Natural Language Processing (NLP) technologies to improve process planning applications [12]. At the May 1996 *International Workshop on Modeling the Product Realization Process: Innovative Methods, Computer Tools, and Applications*, it was discussed that a standard process representation would be useful in applications which track legal processes in the judicial system.

Processes and process information are fundamental to manufacturing. As highlighted above, manufacturing industry has stated the need for formal process representation. Overall consensus on a formal representation would greatly improve its value to industry as a tool for process understanding, development, simulation, control, and integration. A major stumbling block is that no individual company has a vested interest in creating a common, robust process representation beyond their own, specialized needs. Considering NIST's mission to promote U.S. economic growth by working with industry to develop and apply technology and standards, NIST is in an ideal position to facilitate research and development of a formal, standard process representation.

1.2 The SIMA Program

Within the Manufacturing Engineering Laboratory, a major research, development and standardization program was established in 1994, called the Systems Integration for Manufacturing Applications (SIMA) program. Initiated as part of a new federal government initiative on High Performance Computing and Communications (HPCC), SIMA is supporting an expanded program in advanced manufacturing systems integration technologies; development and testing of prototype components and interface specifications for manufacturing systems; application of high performance computing and networking technologies to integrate design, planning and production processes; and testbeds for achieving cost-effective application of advanced manufacturing systems and networks.

Projects within the SIMA program address a variety of technical issues related to manufacturing systems integration. Some projects focus on the development and prototyping of integrated systems, others on the definition of new interface and communication specifications, and yet others on facilitating the standardization process itself in the domain of manufacturing system interfaces and data models. To better understand and manage these various stages and kinds of activities, the Manufacturing Systems Integration Division has developed the notion of an Initial Manufacturing Exchange Specification (IMES) [42]. An IMES represents an interface or exchange specification developed at NIST which at maturity is ready for submission to a standards body as a candidate standard. An IMES has several stages of development, including identification and definition of industry need, requirements analysis, design and development of the specification, validation, consensus building, technology transfer, and standardization. SIMA projects are at various stages in this process. The project under discussion in this paper, the Process Specification Language Project, is at the stage of requirements gathering and analysis.

1.3 The Process Specification Language (PSL) Project

The goal of this project is to identify or create a process specification language (PSL) that can be common to all manufacturing applications, generic enough to be decoupled from any given application, and robust enough to be able to represent the necessary process information for any given application. Additionally, the PSL should be sufficiently well-defined to ensure complete and correct exchange of process information among established applications. This PSL would facilitate communication between the various applications because they would all "speak the same language", either as their "native" language or a "second" language, for exchange.

It is our hope that a lasting solution will be found which will not suffer from a single limited perspective, and will lend itself to standardization. Unlike many computerized languages in use today, we further feel that this language will benefit from a formally described, notation-independent information schema underlying all of the possible notations which might arise. By formalizing the structure of the language in this way it becomes possible to use multiple alternative notations to convey the same information, therefore enabling multiple "views." It is reasonable to assume that the language will have at least one computer-interpretable notation – presumably text-based – as well as several graphical notations which make it easy for humans to create and manipulate the process specification. It is likely that no single graphical notation will be sufficient to convey all aspects of a process, providing instead a particular view of the process. Several such graphical notations used together would provide a comprehensive view. This is similar to the use of various charts and displays in project management systems today, such as a Gantt chart, PERT chart, and Work Breakdown Structure.

The approach taken for the project has been to break it into distinct phases, namely: requirements gathering, existing process representation analysis, schema definition, language grammar/syntax development, language notation(s) development, pilot implementation and validation, and finally, submission as a candidate standard. Each of the IMES stages discussed in Section 1.2 are incorporated into these activities. The second phase, the representation analysis, is designed to determine how well existing representations support the requirements found in phase 1. This analysis will provide an objective basis on which to identify the representation or combination of representations that provides the best coverage of the requirements and to identify gaps in existing representations' abilities to address process specification requirements. The language schema, grammar, syntax, and notation will be developed as a result of this analysis. By this time, a suitable scenario or group of scenarios will have been identified for a prototype implementation to ensure the completeness and ease-of-use of the specification language. It is this validated, documented language that will be submitted to appropriate organizations as a candidate standard. Feedback and consensus from the process community will be and has been aggressively pursued during all phases of the project. The anticipated timeframe for the phases is shown in Figure 2.

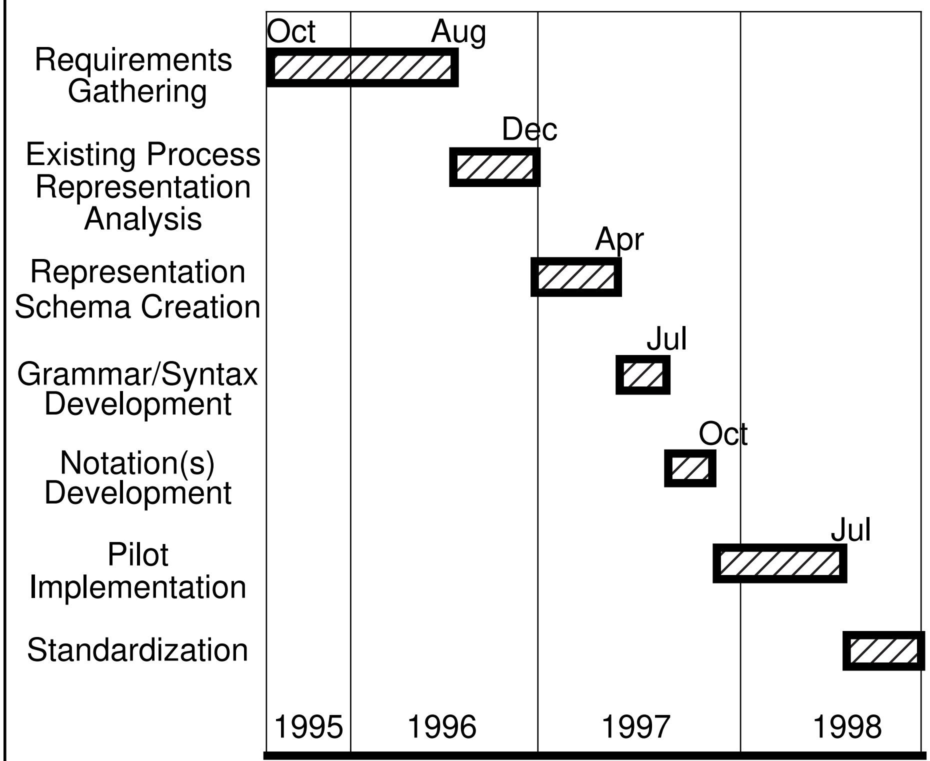

Figure 2: Time Table

1.4 Project Scope

To keep this work feasible, the scope of study is limited to the realm of discrete processes related to manufacturing, including all processes in the design/manufacturing life cycle. Business processes and manufacturing engineering processes are included in this set of requirements both to ascertain common aspects for process specification and to acknowledge the current and future integration of business and engineering functions.

While the intent of phase one of the PSL project is to gather as comprehensive a list of requirements for specifying processes as is possible, the PSL developed by the completion of this project will not address all of these requirements. The PSL will focus on those requirements determined to be most germane to process specification and to those requirements pertinent to the implementation scenario. Additionally, the requirements list is meant to show the types of information that the PSL must be able to access — either by internal representation or by links to other models which include that information. Only the requirements that fall within the scope of the project that are not represented elsewhere will be modeled in the specification language. All requirements represented externally will only be referenced from its source and not remodeled. The requirements listed in Section 2 that are expected to be referenced in other models are appropriately footnoted.

An instance of the process specification language would be "the glue" that relates the data derived from a number of other models as shown in Figure 3. A listing of these additional models along with their respective definitions are described below:

- **Business Practices Model** - the rules, strategies, and policies of a business. This may include information such as safety regulations and quality specifications.
- **Forecast and Customer Orders Model** - the expected and actual orders for a product.
- **Inventory Model** - the current or projected amount of a resource available.
- **Manufacturing Rules Model** - the accepted practices for manufacturing operations. This may include information such as standards data and MIL specs.
- **Process Characterization Model** - a model describing the behavior and capabilities of a process independent of any specific application.
- **Product Specification Model** - the behavior and capabilities of a product (e.g. parts of STEP).
- **Resource Characterization Model** - the behavior and capabilities of a resource.

For example, an instance of the process specification language may include pointers to equipment information from a resource characterization model, product configurations from a product specification model, shop-floor process capability information from a process characterization model, standards requirements from a manufacturing rules model, inventory levels and expected shipment dates from an inventory model, an estimation of the expected number of orders from a forecast and customer orders model, and business guidelines on how to manufacture a product from a business practices model.

It is also worth mentioning that our definition of process is not limited only to machining processes. Examples of other types of processes that fall within scope are intertask processes such as transportation and machine setup, that is, processes which do not provide added value to a product. Also assembly processes and business processes such as approvals are included.

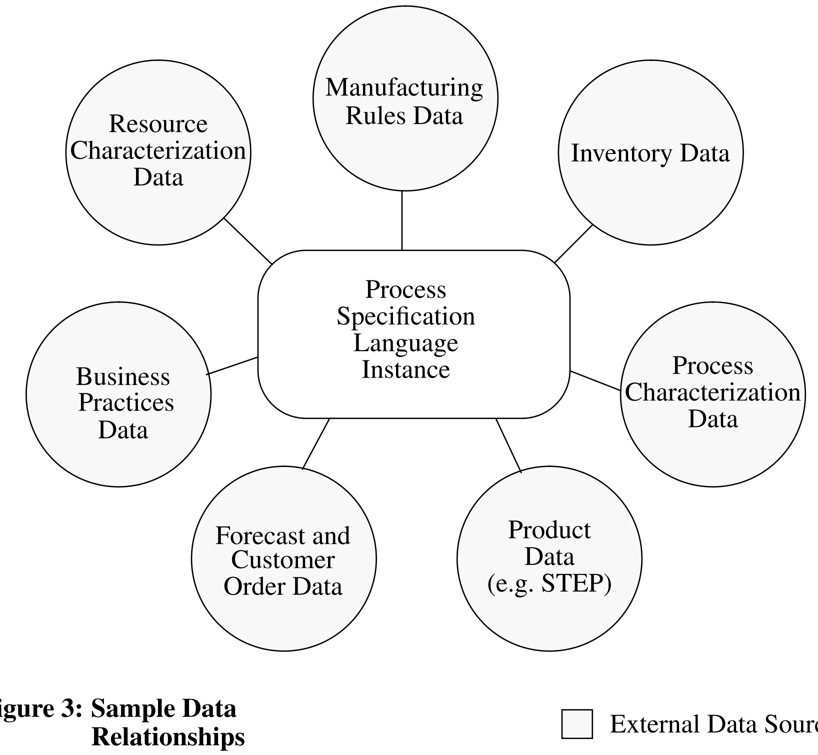

Figure 3: Sample Data Relationships

The terms “process” and “process model” are interpreted in very different ways by different readers. It is therefore appropriate to clarify what we mean by such terms. The goal of this project is to create a process specification language, not a process characterization model. Our definitions of each follow:

- **Process Specification Language**: a language with which to specify a process or a flow of processes, including supporting parameters and settings. This may be done for prescriptive or descriptive purposes and is composed of a schema, a grammar, and one or more notations.
- **Process Characterization Model**: a model describing the behavior and capabilities of a process independent of any specific application. An example would be a numerical model capturing the dynamic behavior of a process or limits on the process' performance or applicability.

The relationship between the concepts of a characterization model and a specification language is similar to the relationship between an English language dictionary and a specification of the usage rules of the English language. While a dictionary defines the words in the English language, a usage rule specification states how these words can be combined into sentences, etc. Analogous to this, a Process Characterization Model may define processes while a Process Specification Language may describe their combination into sequences of actions, etc. To carry the analogy further, an instance of the Process Specification Language is similar to a sentence or a paragraph in the English language. In both cases, definitions and usage rules are combined to communicate a specific concept.

1.5 Purpose of this Document

This document represents the findings of the requirements gathering phase of the PSL project, wherein requirements have been collected from a number of applications, then categorized and classified. These categorized requirements will provide the foundation for an objective and critical analysis of current process representation methodologies, which is the next phase of the project.

The intent is to create a strong foundation of the requirements necessary for representing process rather than to exhaustively characterize all requirements necessary for each individual application. There are many reasons why it would not be conducive to do so. With the advent of new technologies, these application requirements may be constantly changing, prohibiting us from capturing all of the future requirements that would be necessary to represent process. Because of this, a mechanism was put in place to allow any additional requirements to be added when and where appropriate, namely the extensibility of the model.

Section 2 includes a description of our approach for gathering the process requirements and the definition of the categories in which the requirements are grouped.

Section 3 lists the categorized requirements along with their respective definitions, and a discussion of which requirements are not included and why.

Section 4 discusses the seven applications explored and includes: a brief overview of each application; a description of the current practices for process representation; and a list of the respective process requirements. Also, additional application-specific requirement explanations are included where appropriate.

Section 5 summarizes the findings and describes the next phases in the project.

The intended audience of this paper is the technical community of manufacturers, software vendors, researchers, and standards bodies who might have an interest in the robust and satisfactory performance of such a common representation.

The paper is designed for two types of readers: application experts who wish to review what requirements are necessary to represent process in their application, and researchers and practitioners who have an interest in process representation independent of any one particular application. For application experts, the sections of most relevance are the Introduction, Sections 2 and 3 (Requirements for Process Representation), the appropriate application(s) in Section 4 (Application Requirements), and Section 5 (Conclusion). For researchers and practitioners who have a more application independent interest in process representation, the sections of most relevance are the Introduction, Sections 2 (Requirements for Process Representation), and Section 5 (Conclusion); Section 4 (Application Requirements) can be skimmed or skipped altogether.

2.0 Requirements for Process Representation

2.1 Approach

In order to obtain a comprehensive list of requirements needed to represent process, several avenues were pursued. The first was to explore a cross section of applications which use process information to determine the process requirements for each. This was primarily accomplished through a literature survey and by talking to application experts. The applications that were explored, in no particular order, were process planning, production scheduling, simulation, workflow, business process re-engineering, project management, and product realization process modeling. It should be noted that although simulation is actually a technique instead of an application, for the purposes of this project, it will be treated as an application because it makes apparent its own set of requirements in much the same way as the other applications explored.

Another avenue for gathering process requirements was talking with researchers who have gathered similar requirements to represent process but limited their scope to particular industries. For example, the semiconductor industry has done substantial research in the area of process representation for wafer fabrication.

Existing software packages, modeling languages, architectures, and standards were also examined. Again, only a subset of these were selected due to limited resources and time constraints. Some of the efforts explored were ALPS (A Language for Process Specification) [9], MOSES (Model Oriented Simultaneous Engineering Systems) [49], BPFL (Berkeley's Process Flow Language), CS/CAPP™ process planning system [16], AP213 (an application protocol in STEP) [38], MetCAPP™ [49], Workflow Management Coalition (WfMC) specifications [71], Knowledge Interchange Format (KIF) [26], Microsoft Project ™ [48] and SDEF (Standard Data Exchange Format) [20].

In an attempt to group these requirements in a logical, cohesive fashion, four categories emerged and have been named Core, Outer Core, Extension and Application. Each of these categories is explained in Section 2.2. The categorized requirements were matched to the various application requirements discussed above, to show specific examples of how each can be tailored toward specific applications.

To ensure that the process model is truly robust, active participation of practitioners and researchers has been encouraged. Two mechanisms were put in place to allow these colleagues to be kept informed and to actively participate in the PSL project.

The first was the creation of a series of HTML pages made available through the World Wide Web (http://www.nist.gov/psl/). These pages were updated weekly to reflect our current status during the requirements gathering phase. Linked to these pages are feedback forms to allow readers to provide comments and suggestions.

The second mechanism was the creation of two mailing lists set up for the purpose of informing colleagues about the status of our project in a more proactive fashion. The first list was created to allow one way communication from the project participants to interested parties to notify them when milestones in the project were reached. This usually involved pointing the community to the web pages when a substantial change was made. This list currently includes over 200 participants and continues to grow. The second list was created to allow for discussion among the project participants and the external community. Through communication and external feedback, the scope and direction of the project has been and will continue to be tailored to better coincide with the expectations and desires of the community. This second list currently includes over 40 participants and is also growing.

2.2 Category Definitions

In an attempt to group the requirements for the process specification language in a logical, cohesive fashion, four categories emerged. Each category was further subdivided into representational and functional requirements. This grouping is intended to improve readability and understanding of the requirements rather than to define the architecture for a future representation. Definitions and examples of this grouping are described below.

TABLE 1. **Requirements Categorization**

	Representational Requirements	Functional Requirements
Core	e.g. resource, task	e.g. extensibility
Outer Core	e.g. conditional task	e.g. exception handling
Extensions	e.g. process performance measurements (Analysis)	e.g. resource monitoring and feedback (Analysis)
Application-Specific	e.g. non-machining times (Production Scheduling)	e.g. dynamic rescheduling (Production Scheduling)

2.2.1 Definitions

- **Core:** the most basic, essential requirements inherent to all processes. To represent process, it is either critical that these requirements be included, or these requirements are so common that every application either explicitly or implicitly uses them. While all processes contain core requirements, the core requirements provide the basis for representing only the simplest of processes. (e.g. time, resource, activity)

- **Outer Core**: the pervasive, but not essential, requirements for describing processes common to most applications.(e.g. temporal constraints, resource grouping, alternative tasks)
- **Extensions**: the groupings of related requirements, common to some, but not all, applications that together provide an added functionality. Although the requirements listed within the extensions are not inherently necessary for representing process, they are useful during implementation to provide their respective functionality. Six extensions have been defined, each described below:
  1. **Administrative/Business -** requirements supporting the description of the policies, priorities, and rules of a company that could affect processes. Concept such as business practices, manufacturing rules, design reviews, go/no-go decisions, and ISO9000 rules would be supported here.
  2. **Planning/Scheduling/Quality/Analysis** - requirements supporting analysis, quality checking, and process performance measurements.
  3. **Real-Time/Dynamic** - requirements supporting real-time monitoring of a process, such as process/resource status and states.
  4. **Process Intent** - requirements supporting the description of the functions and goals of processes, such as process purpose and decision rationale.
  5. **Aggregate Resources/Processes** - requirements supporting the description of the characteristics of a group of resources. Concepts such as parallelism and implicit resource association are included here.
  6. **Stochastic/Statistics** - requirements supporting the description of the random or probabilistic aspects of a process. Concepts such as probabilities of down time and performance variability are included here.
- **Application Specific**: the requirements only relevant within specific applications (e.g., dynamic rescheduling for the production scheduling application). There were seven applications explored during the requirements gathering phase: process planning; production scheduling; simulation; project management; enterprise re-engineering; workflow; and product realization process modeling. This is not meant to be an exhaustive list of applications which use manufacturing process information. It is only meant to serve as a good sampling to determine a comprehensive set of requirements necessary to represent process information.

Each of the above categories are sub-divided into the following two types of requirements:

- **Representational Requirements**: the characteristics of a process that the process specification language must be able to implicitly or explicitly represent. Some examples are resources, parallel tasks, and non-machining times.
- **Functional Requirements**: the activities or behaviors relating to a process that the process specification language must be able to support. Some examples are exception handling, dispatching, and deadline management.

2.2.2 Practical Example

The different categories of requirements described above are combined when supporting a particular implementation. For example, a Workflow implementation would include all of the requirements in the core, many of the outer core requirements, those extension requirements that are pertinent, and the workflow application requirements, as shown in Figure 4c. Also note that the core, outer core, and extension requirements can be shared by different applications.

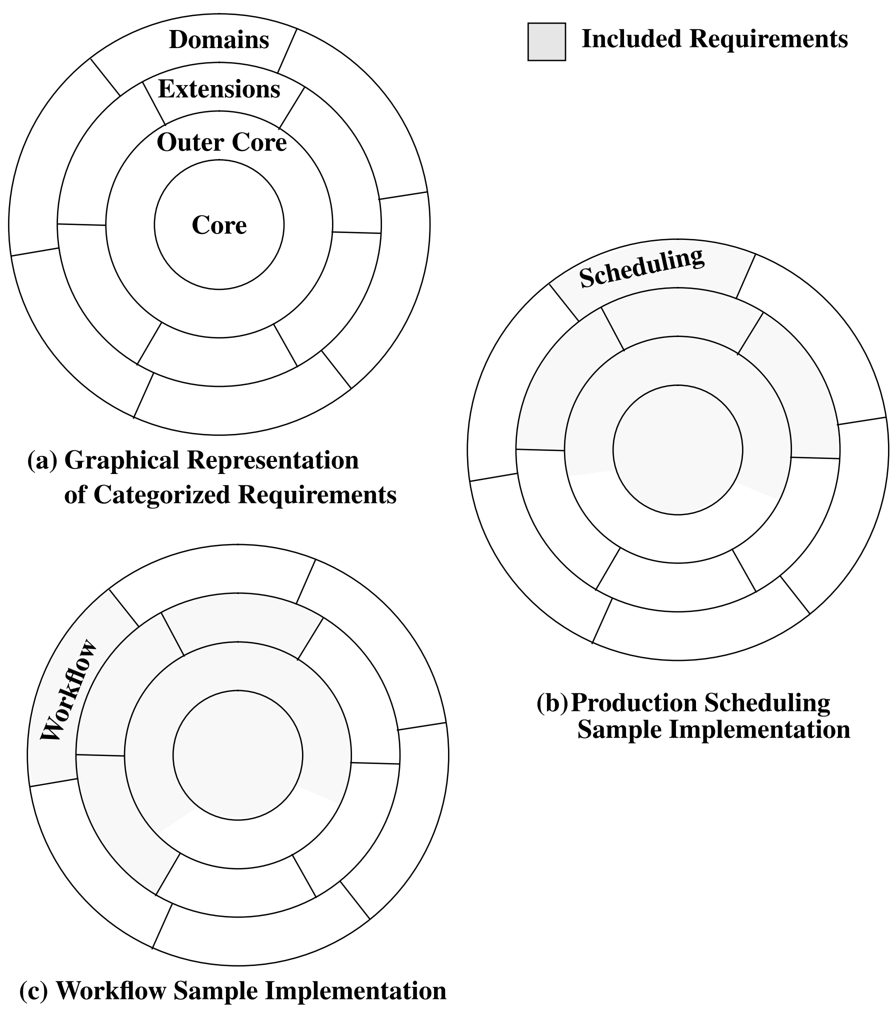

Figure 4: Sample Implementations

3.0 Categories and Associated Requirements

The following is a categorized list of requirements for representing process. The requirements in each category are listed in alphabetical order to remove any implied relative importance. Some requirements may appear redundant, but are included for clarity. Definitions and examples are included next to the respective requirement where appropriate. An explanation of representational and functional requirements can be found in Section 2.2. For a more compact list of categorized requirements without their respective definitions and examples, please refer to Appendix A.

3.1 Core Requirements

Core requirements, by definition, are pertinent to all applications which use a process representation.

3.1.1 Representational Requirements

1. ad hoc notes and annotations optionally associated with any component of a plan - on-the-fly, off-the-cuff notes and documentation. This could be voice, video, as well as text. A person's observation of a process might go here.
2. cost data - the cost associated with a resource or task. This could be a fixed cost, cost rate, or a cost derived from other attributes such as duration and level of effort. Costs associated with uncertainty, variability, tolerances, etc. could also be included in this requirement.
3. level of effort - description of the amount of a resource needed, in any given unit, to accomplish a task. Some example levels of effort are equipment-hour, labor-hour, and crew size.
4. product (work item) characteristics[1]- information about an intermediate and final product which a process will produce.
5. resource[1] - a single resource or a group of resources. Some types of resources are equipment, people, information, and in-progress goods.
6. resource requirement(s) for a task (with quantity) - the relationship between one or more resources and a task. For example, a drilling task may use drilling machine A, drill bit B, fixture C, and coolant D. The resource may play one of many roles including agent, operand, ancillary resource, etc.

1. All future footnote references in this section will refer to this footnote. This footnote is also repeated at the end of this section. Because past and/or present efforts have already attempted to model this type of information, this work will reference the models that are the result of those efforts instead of duplicating the information.

7. simple groups of tasks - very basic, high-level set of tasks. One example is the grouping of tasks and sequences that make up a process plan or that make up a phase.
8. simple resource capability/characteristics[1] - a high-level description of the characteristics of a resource. More detailed descriptions can be found in the outer core.
9. simple sequences - linear, time-sequential groups of tasks. More sophisticated relationships such as parallel and alternative tasks can be found in the outer core.
10. simple task representation and characteristics - a simple, high-level description of the task. More detailed representations can be found in the outer core.
11. task duration - the time required to complete a task or group of tasks. Only simple durations are represented here. Other types of associated times, such as estimated duration, actual duration and theoretical durations, are included in the outer core.
12. task executor - who is responsible for executing a task or group of tasks. Examples include a person, controller, or external company if the task is contracted out.

3.1.2 Functional Requirements

1. extensibility - there must be a mechanism in place to allow a user to add additional information to the pre-defined data constructs. One such mechanism could be the addition of stubs for user-defined information.
2. resource allocation/deallocation for one or many tasks - the assignment and release of one or more resources to a task of group of tasks.
3. simple precedence - a high-level description of the precedence constraints of one task on another. A more detailed description of precedence and constraints can be found in the outer core.

3.2 Outer Core Requirements

Outer core requirements are pertinent to most applications which use a process representation. Listed after each requirement definition are some sample applications which would need to represent this requirement.

3.2.1 Representational Requirements

1. abstraction - within the scope of this project, there are three concepts of abstraction that must be captured. The first is the description of information at various, hierarchal levels. This could include the composition and decomposition of process information. For example, one may describe a process at a high level of abstraction by saying “a task is to be performed on a part.” At a lower level, one may describe the same task by saying part 321 needs to be have an operation performed on it by resource 123 using the resource ABC at 12 p.m. The second concept is the idea of

incompleteness. When not all of the specific information about a process is known, the representation needs to be able to accept that and perform normally until the time that the information is required. For example, one might know the resource in which a specific part is to be manufactured, but not know the attributes that are associated with it. That additional information may only be discovered just before the task occurs. The third concept, which is closely related to the incompleteness concept, is that of ambiguity or vagueness. An example of vagueness is when a manufacturer specified that an operation needs to be performed on an in-progress part. There may be many options to do this, but the manufacturer purposely does not go into detail because this decision has not yet been made.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

2. alternative task - (see complex sequences)

3. associated illustrations and drawings - aural or visual information associated with a resource or task to provide a clearer explanation on how to perform the task.

[Production Scheduling] [Process Planning] [Product Realization Process Modeling]

4. complex groups of tasks - groups of tasks which have a common tie. One example of such a grouping is the creation of a job consisting of all the activities that are performed on a given resource. Another example is the ability to assign resources (e.g. costs) to "work packages", which may consist of numerous tasks without individually allocated resources (Simple groups of tasks are found in the core requirements.)

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

5. complex resource characteristics[1] - a detailed description of the characteristics of a resource or group of resources. One such characteristic may be the ability of a resource to provide multiple functions. Also, information about the type of resource, such as non-replacable or expandable, could be included. Simple resource characteristics can be found in the Core.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

6. complex sequences - complex ordering relationships between tasks. Some examples are listed below:

alternative task(s) - one or more tasks that could perform that same function as a specified task. Alternative tasks can help with optimization.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

concurrent tasks - two or more tasks that must occur at the same time.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

parallel tasks - two or more tasks that occur at any time with respect to one another.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

serial tasks - two or more tasks that must occur one after another.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

7. complex task representation and parameters[1] - a detailed representation of a task or group of tasks. This could include dummy tasks and non-value added tasks such as transport, set-up, and maintenance. One such task characteristic is that of uninterruptability. Other information that can be included is a description of the behavior and capabilities of a process independent of any specific implementation, such as limits on the process' performance or applicability.Simple task representation can be found in the Core.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

8. concurrent tasks - (see complex sequences)

9. conditional tasks - a task that only needs to be performed under some predefined circumstance. For example, a given task may only need to be performed if an attribute is greater than a certain amount.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

10. confidence levels - a measure of certainty that some attribute is true. For example, there may be a high degree of confidence that a task will be completed within 20 minutes.

[Production Scheduling] [Process Planning] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

11. constraints[1] - implicit or explicit constraints associated with a task or resource. Some examples are listed below:

temporal constraints - the time-related characteristics or relationships relating to one or more tasks. This would include, but not limited to, the well-defined 13 time interval relationships.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Project Management] [Product Realization Process Modeling]

pre- and post-conditions constraints - conditions that must be satisfied at the beginning or end of a task.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Project Management] [Product Realization Process Modeling]

state existence constraints - requirement(s) on the existence of a state in order for a task to execute.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Project Management] [Product Realization Process Modeling]

material constraints - this might include information such as a material's withstandable force.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Project Management] [Product Realization Process Modeling]

12. date(s) and time(s) and/or multiple duration(s) - the association of one or more dates and times and/or multiple durations with a resource or task. For example, it may be important to know when an event started and ended by attaching two dates and times to each event. It may also be important to represent different types of durations with a single task as well as required delays between tasks. For example, a task might have an estimated duration, an actual duration, and an average duration.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

13. implicit/explicit resource association[1] - an implicit or explicit dependency of a resource on another type of resource. For example, a resource may need another resource in order to work properly. One might never care exactly what additional resource is used, only that one existed and was used.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Project Management]

14. iterative loops - a situation when a task or group of tasks repeats until a desired condition is met. For example, if you want a product's attribute to be within a certain measurement, you may have to perform a task an unknown amount of times until that condition is met.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Product Realization Process Modeling]

15. manual vs. automated tasks - characteristics of a task can differ depending on if a human or a machine is performing that task. This may also be true of two different humans or two different machines.

[Production Scheduling] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

16. manufacturing product quantity - the amount of the product that is to manufactured. This can play an important role in deciding what processes and resources are to be used. Usually the lower the batch size, the more specialized the product.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow]

17. material constraints - (see constraints)

18. parallel tasks - (see complex sequences)

19. parameters and variables - place holders that can store a constantly changing value. They are important for making processing decisions from real-time data. For example, task A may only need to be performed when a specification is outside the allowable boundaries. A variable can be passed to task A with the specification's value to decide if it needs to be performed.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Product Realization Process Modeling]

20. pre- and post-processing constraints - (see constraints)

21. queues, stacks, lists - the representation of an ordered or unordered group. Such a group may be used to represent a resource's input queue.

[Production Scheduling] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Product Realization Process Modeling]

22. resource categorization and grouping - a logical grouping of resources with a common tie. Some group characteristics may be a resource's function and/or location.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

23. resource location.[1] - identification of the location of a resource. This could be useful for resource layout optimization

[Production Scheduling] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Product Realization Process Modeling]

24. resource/task combined characteristics - qualities of a resource that are dependent on a particular task, or qualities of a task that are dependent on a particular resource. An example of the former is when Resource A is performing task B, it must not be preempted by any other task. An example of the latter is various resources may take different times to perform the same tasks. Therefore, the task time is dependent on the resource selected.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

25. serial tasks - (see complex sequences)

26. state existence constraints - (see constraints)

27. state representations - the description of a process in terms of any combination of the states of the process and/or resource.

[Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

28. temporal constraints - (see constraints)

29. uncertainty/variability/tolerance - the representation of the deviation from the nominal. A task parameter that may have tolerances associated with it is time.

[Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

3.2.2 Functional Requirements

1. ability to insert or attach a highlight (milestones) - the ability for a user to highlight a section of the process. This might entail highlighting an existing task or adding a new, zero-duration task such as a dummy task to show the importance of an aspect of a process. An example of a highlight might be the addition of a milestone.

[Production Scheduling] [Process Planning] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

2. complex precedence - the ability to convey a series of tasks' ordering requirements within a given process. Some examples of complex precedence constraints are: event A occurs before event B if event C occurred within 3 hours; or multiple inputs must all occur before a task can begin. Conditional precedence could also be included here. Simple precedence constraints are found in the core.

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

3. convey the ancestry or class of a task - the ability to describe a task as it relates to the specialization of another, higher-level task. For example, a low-level task A may be a specialization of a higher-level task B which may be the specialization of an even higher-level task C. In this sense, characteristics of task C would always apply to a task B and characteristics of a task B would always apply to task A.

> [Process Planning] [Enterprise Engineering and Business Process Re-engineering]

4. deadline management - the ability to consider a predetermined deadline when making decisions. For example, the decision as to whether or not to use a given task could be dependent on the timeframe in which work has to be completed and at which point in the timeframe the activity is currently in.

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

5. dispatching - the determination and representation of rules and guidelines to decide when items should be released for production. This could be used for line balancing, reduced buffer size, Just-In-Time production, etc.

> [Production Scheduling] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Product Realization Process Modeling]

6. eligible resources - the ability to determine which resources can be chosen for a task (selection rules). In many implementations, these resources are implicitly or explicitly grouped together and the appropriate resource is chosen by considering its availability and expected performance measures.

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

7. exception handling and recovery - the ability to specify corrective action when a task fails. Some actions may include using an alternative task or aborting the process.

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Product Realization Process Modeling]

8. information exchange between tasks - the ability to represent the flow of information among tasks. This information may be used to make decisions about upcoming tasks. One such way of accomplishing this is through parameter passing.

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Product Realization Process Modeling]

9. mathematical and logical operations - the language must be able to perform mathematical and logical operations. This could help to derive values from existing data, help with decision making, and facilitate branching (AND/OR splits, etc.).

> [Production Scheduling] [Process Planning] [Simulation] [Work Flow] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

10. support for task/process templates - the language must allow for templates of a task or process. These templates might include the types of information that a process needs to know but not the values that go along with them. It is the user's or program's responsibility to determine and populate the values. Templates could provide a good mechanism for re-using process elements.

    [Production Scheduling] [Process Planning] [Work Flow] [Project Management] [Product Realization Process Modeling]

11. support for simultaneously maintained associations of multiple levels of abstraction - the ability to associate information at multiple levels of abstraction with a task. For example, one may want to associate a resource with a task. At the beginning, one may only say that five people are needed to perform a task, later one may specify five people with a special ability, and later one may want to explicitly say which five people they are. It may be important to capture all of these levels of abstraction together so that if one of the people is unable to perform their duty, it will be easy to see why that person was chosen and another can be easily selected.

    [Production Scheduling] [Process Planning] [Enterprise Engineering and Business Process Re-engineering] [Project Management] [Product Realization Process Modeling]

12. synchronization of multiple, parallel task sequences - the ability to specify a mechanism to coordinate two or more tasks that occur at the same time. One possible mechanism to accomplish this is event signalling and notification.

    [Production Scheduling] [Process Planning] [Simulation] [Enterprise Engineering and Business Process Re-engineering] [Work Flow] [Project Management] [Product Realization Process Modeling]

3.3 Extension Requirements

The extensions listed below are groups of requirements that together provide some added functionality. Any application that wishes to provide a given functionality would include the respective extension.

3.3.1 Administrative/Business

3.3.1.1 Representational Requirements

1. business practices, rules, constraints[1] - the practices and policies of a business that may relate to a process. For example, a policy of "Only use resource X for one hour at a time because it is expensive to run" may cause a person to plan around it. Other types of data that may be included in this category are hazards and security recommendations, safety regulations, business policies, manufacturing rules, and quality specifications.
2. configuration management information and processes - Some examples of configuration management types of data and processes are: approvals, archiving, contracts,

effectivity, organizations, process logging, security classification, time zones, and version control.

3. customer-driven process specification and constraints[1]- the specifications by a customer or external organization on how a product is to be made. The could include either customer requests or standards information such as MIL specs.
4. forecast and customer orders[1] - the expected and actual orders in some time period.
5. priority attributes - the relative importance associated with a particular characteristic of a resource or a task. This may change from business to business and from task to task.
6. representation of the origin of task(s) - a description of where a task or group of tasks were originally obtained. Some example origins are libraries of previously used tasks and newly created tasks.

3.3.2 Planning/Scheduling/Quality/Analysis

3.3.2.1 Representational Requirements

1. analysis characteristics - the representation must have constructs available to describe the results of analyses of processes. Some types of analyses (in no particular order) are cost, quality, process duration (time [theoretical, experimental, estimated]), task sensitivity, safety, manufacturability, line balance, critical path, likelihood of error, resource utilization, throughput and volume, quality of service, effort, queue time, queue length, driving resources, risk, complexity metrics of groups of tasks, lag time, float, expected/actual process yield, percent idle time and busy time for a resource, mean time between fails/mean time to repair, and defects per unit (DPU).
2. critical task - a task which has been determined to lie along the critical path. *(see glossary)*
3. predictive and time-dependent resource availability - a preliminary guess as to whether a resource will be available at a future time. This could be used to determine which type of resource to recommend to perform a task. It is also possible that a resource is not available at a certain time, therefore can not be considered to perform a task.
4. prescriptive task behavior - a preliminary guess as to how the task will behave. This could include task duration and expected output.
5. task/process performance measurement - a measurement and/or description of the quality of a task, process, or resource. This requirements is also listed in the Real Time/Dynamic Extension.

3.3.2.2 Functional Requirements

1. co-existence of plans and resolution of conflicts - there must be a mechanism, specification, or algorithm in place to resolve conflicts when two or more existing plans use the same resource at the same time.
2. dynamic model modification - the ability to dynamically change the process model in order to improve some aspect of the process. This requirement is also included in the Real Time/Dynamic Extension.
3. optimization - the ability to perform analysis to determine the optimal solution for a given problem. This could include the optimization of any of the analysis characteristics in the representational requirements section of this extension.
4. resource/system/process monitoring and feedback - the ability to observe and regulate a resource, system, or process over time. This could be used to adjust various parameters to improve quality. This requirement is also included in the Real Time/Dynamic Extension.
5. support for validation of the entire process plan - the ability to allow the user to verify that the process plan will perform as expected.
6. tracking of changes in the system - the ability to determine when a system changes. This could cause one to find an appropriate way to work around the change to accomplish a predetermined goal. Some possible changes are equipment breakdowns and additions of new or upgraded equipment.
7. what-if analysis - the ability to do analysis on a task or process by adjusting parameters to see what would happen if those parameters were to occur in the future. An example might be the determination of the effect on a process if an attribute in a resource were to deviate by 10%. What-if analysis can be performed for all of the types of analysis mentioned in the representational requirements in this extension.

3.3.3 Real-Time/Dynamic

3.3.3.1 Representational Requirements

1. resource amount and availability - the availability of a resource at any given time. Some reasons a resource may not be available are: it is performing another operation; it is undergoing maintenance; or it is broken. Because the availability changes, this determination should be done in real-time.
2. resource interruptions - constructs to capture when a resource is halted at an unexpected time. Some reasons for this may be equipment breakdown, loss of power, or user error.
3. process yield - the actual output and performance of a process. This could be collected in real-time to perform analysis or for post-processing.

3.3.3.2 Functional Requirements

1. dynamic model modification - the ability to dynamically change the process model in order to improve the output's quality. This requirement is also included in the Planning/Scheduling Quality/Analysis Extension.
2. event signalling and notification - the ability to alert a user or another task when an activity reaches a predetermined state or stage. This could be used for synchronization of multiple parallel tasks (found in the outer core).
3. resource behavior during processing time - the ability to describe the behavior of a resource over time. An example would be a resource's degradation during processing time.
4. resource/system/process monitoring and feedback - the ability to observe and regulate a resource, system, or process over time. This can be done in real-time to dynamically make modifications to an existing process. This requirement is also included in the Planning/Scheduling Quality/Analysis Extension.
5. tracking of changes in the system - the ability to tell when a system changes. This could cause one to find an appropriate way to work around the change to accomplish a predetermined goal. Some possible changes are equipment breakdowns and additions of new or upgraded equipment. This requirement is also included in the Planning/Scheduling Quality/Analysis Extension.
6. track in-progress goods - the ability to tell, at any given time, the state of a product that is to be manufactured.

3.3.4 Process Intent[1]

3.3.4.1 Representational Requirements

1. decision rationale - the reason behind a decision. This could help in the future if the original decision no longer holds and an alternate decision must be made.
2. intentional dimension of processes, or goals - an explanation of the desired goals of a process. The goals may be explicit, such as "bring the product's attribute down to a certain level", or vague, such as "improve the quality of this product."
3. relationship between task and goal and resource and goal - have constructs available to associate a specific goal with resources and tasks that are able to fulfill that goal. This association can be decomposable and traceable down to every last task in the process at the lowest level of abstraction.
4. task/process purpose - the task's or process' function within a specified plan. For example, the creation of a product's attribute as shown in a diagram may be the intent behind a given operation.
5. value-added attributes - information and/or parameters that are associated with some aspect of a task or process whose purpose is to add more insight into its purpose.

3.3.4.2 Functional Requirements

1. access to past and present decision rationales - the ability to find and retrieve information describing why past process decisions were made. This could help to decide if a previous process can be used for a current goal and if a modification to an existing process will provide the output needed.

3.3.5 Aggregate Resources/Processes

3.3.5.1 Representational Requirements

1. characteristics of groups of resources - a resource within a group may have different characteristics than if it were alone. For example, if it is possible that two resources can share the same physical space, only one resource can occupy that space at any given time.
2. implicit task association - the association and dependency of a task on another task. For example, you may need to always perform a preliminary task on a piece of equipment before you can use it to perform a given task.
3. parallelism - the representation of a resource that can perform multiple, simultaneous functions at once. Because these resources have their own set of rules and constraints, they must be modeled as an aggregate resource. A person can be considered an aggregate resource since he/she can possibly perform multiple functions at once.

3.3.6 Stochastic/Statistics

3.3.6.1 Representational Requirements

1. descriptive manufacturing/performance variability - the variability of a task's or resource's performance. This could be from historic data, a resource handbook, or other source. Confidence levels could be derived from this information.
2. probabilities of down times - a statistical approximation of the probability of a resource not functioning correctly. Some reasons for this could be breakdown, incorrect settings, and need for calibration.
3. stochastic properties - attributes which involve a degree of probability or chance. Some examples, in no particular order, are: process yield; feed qualities; operating conditions; equipment failure rates; rejection rates at test stations; equipment jams; station cycle time; and part outages.
4. uncertainty of sequences - a measure of doubt about the ordering of tasks. When multiple paths can be chosen to accomplish a goal, there is no sure way to determine exactly which will be chosen. The choice may depend on resource availability,

time constraints, etc. and may only be determined in real-time. The nature and extent of the uncertainty may change as the process matures.

3.3.6.2 Functional Requirements

1. account for randomness - the ability to consider events that do not have predetermined or computable attributes. For events that can only be described as happening in a random fashion, a mechanism must be in place to somehow quantify the data. This can usually be accomplished by some kind of probability distribution on historical data.
2. stochastic functionality - the language must be able to handle circumstance when the outcome is not predetermined. It must be able to evaluate different alternatives and assign probabilities, or use other mechanisms, to predict the expected outcome.

3.4 Application Requirements

Application requirements are only applicable to the application in which they are listed.

3.4.1 Production Scheduling

3.4.1.1 Representational Requirements

1. production type data - a description of what type of production mechanism to use when manufacturing a given product. This is usually dependent on the number of products to be produced. Some production types are mass production, job shop, and batch. [73]

3.4.1.2 Functional Requirements

1. dynamic rescheduling - the ability to reschedule a process in real time. This may be done in response to machine down times or lack of resources.[6]

3.4.2 Process Planning

3.4.2.1 Representational Requirements

1. clamping surfaces - where a fixture can be placed to ensure product specifications are not disturbed. [11]

2. datums and offsets - a datum is a description of a typically imaginary line from which measurements are made. An offset it a distance from a line, which may or may not be the datum. This could be used by a process planner to describe where work is to be performed on a product. [11]
3. features to be machined - some examples of features are holes, rivets, subdivisions of unwanted material, and grouping of surfaces in a part which require similar tasks and tolerances. [30]
4. production type data - a description of what type of production mechanism to use when manufacturing a given product. This is usually dependent on the number of products to be produced. Some production types are mass production, job shop, and batch. [73]

3.4.3 Simulation

3.4.3.1 Representational Requirements

1. queue entry and exit rates - a description of the rate in which products are entering and leaving a queue. This could help to determine the efficiency of the machine and help identify and bottlenecks. [66]

3.4.4 Enterprise Engineering and Business Process Reengineering

3.4.4.1 Representational Requirements

1. conceptual entities, such as ideas or knowledge (e.g., an engineer's knowledge of the result of a test) (this could be a specific case of abstraction, incompleteness, ambiguity, accurate - not precise)

3.4.5 Workflow

3.4.5.1 Representational Requirements

1. manual vs. automatable tasks

3.4.5.2 Functional Requirements

1. invoked tool capability - enables automated processes to be invoked by the workflow management system
2. support specifications of task structure (control flow)

3.4.6 Project Management

3.4.6.1 Representational Requirements

1. work breakdown ids

3.5 Other Requirements Not Previously Mentioned

During the review and categorization of process requirements, it was determined that some of these requirements were not directly needed to model manufacturing processes. The reasons for this varied: some were schema requirements; some were language requirements, and others were implementation requirements. Below are the requirements that were not included in the previous lists, grouped with respect to their reason for being left out.

3.5.1 Partial List of Possible Process Specification Schema Characteristics

(to be revisited in phase two of the project)

1. data captured in an unambiguous representation, and able be updated to suit changes in users requirements [2]
2. easy-to-use, good documentation
3. extensible [8]
4. support inheritance [17]
5. able to model the results from different applications that may vary from one another for the same process [8]
6. no redundant information possibly by using derived attributes
7. support multiple views of data

3.5.2 Partial List of Possible Process Specification Language Requirements

(to be revisited in phase three of the project)

1. an instance of the language must be human-readable and computer-interpretable
2. easy access to and encoding of ancillary information and capture of information transactions - the language must have constructs in place to easily access outside information, including information from other databases.
3. must be able to handle binding and scoping
4. work equally well for simple and complex circumstances - this language must provide easy use for those who wish to only include a high-level non-detailed representation of their process, yet it must be robust and complete enough to allow for those who wish to provide a more detailed representation of their process.

3.5.3 Partial List of Possible Implementation Requirements

(an implementation that uses the process specification language should have the following characteristics)

1. allow for concurrency control among simultaneous users
2. facilitate incremental changes to processes
3. intuitive for simple and complex circumstances
4. allow for user administration (identification and management of user/role association)

3.6 Endnotes

1. Because past and/or present efforts have already attempted to model this type of information, this work will reference the models that are the result of those efforts instead of duplicating the information.

4.0 Application Requirements

This section contains requirements that were discovered while exploring a cross- section of applications that could benefit from a common manufacturing process representation. The application requirement listed are not meant to be exhaustive, they are only to show the types of requirements that each respective application would need to represent. The intent is that through an analysis of the requirements for representing both manufacturing engineering and manufacturing business processes, a comprehensive group of requirements would be collected, as shown in Section 2.

The applications explored, in no particular order, were production scheduling, process planning, simulation, workflow, business process re-engineering, project management, and product realization process modeling. Each section follows the same structure: a brief overview of the application; a brief description of current practices; and a list of the types of process requirements that apply to the application with additional description where appropriate. The grouping of the requirements within the applications match the categorized list in Section 2 and are not meant to represent their relative importance within the application. Please refer to the citations for more in-depth coverage of application descriptions, current practices, and requirements definitions.

4.1 Production Scheduling

4.1.1 Overview

Production scheduling is the process of assigning activities to resources over time [3]. It is a decision making process where two kinds of decisions may be taken: 1) Time placement decisions: when should each task start? and 2) Resource allocation decisions: Which resource(s), over time, should each task be executed with? [35] Some examples of resources are machines, operators, cutting tools, fixtures, raw materials, and information.

Production scheduling may aim to produce a variety of parts in line with customer demands and limited resources [55]. The scheduling task is usually based on the material requirements plan (MRP), the master production schedule (MPS), and the capacity requirements plan (CRP).

One may also see production scheduling as a reality check on the planning process, the objective being the implementation of the plan, subject to the variability and constraints that occurs in the real world [6]. Some constraints that may affect scheduling are a task's duration, release and due dates, precedence constraints, transfer and set-up times, resource availability, and resource sharing. Scheduling constraints can be described in one of the following four ways:

1. temporal - precedence relationships between activities

2. resource availability - conditions under which a resource can be made available for use
3. resource utilization - how activities can use and share resources
4. contigencies - degraded modes of operation due to foreseeable external events and less than optimal inputs

Computing a schedule that observes the constraints and objectives of a given scheduling problem is very combinatorial: many alternatives need to be explored, many (inter-wined) decisions need to be made and undone before a feasible schedule can be found. As a result, usually only schedules that are good (as oppose to optimal) are selected due to time constraints.

Although scheduling problems and needs may differ between implementations within the manufacturing application, the basic requirements that are needed to represent them remains fairly constant.

4.1.2 Current Practices

In general, current production scheduling methods used, which are mainly based around the use of due dates, are usually not sufficiently robust for their production environment. Uncertainty in external factors, such as the suppliers, are particularly identified as problems areas. There is a high degree of mismatch between the computer production planning and the control systems being used and the actual product process. The result is that many of the operational decisions still appear to be made manually and without any computer support.[29]

Because of the above, production scheduling is performed under fixed process assumptions and without regard to the opportunities that process alternatives can provide for acceleration of production flows. Only under extreme and ad hoc circumstances (e.g. under pressure from shop floor expediting of late orders), are the process-planning alternatives revisited. This lack of coordination leads to unnecessarily long order lead times, increased production costs and inefficiencies.[58]

Another common problem with scheduling today is the lack of integration with the overall manufacturing system. A scheduling system must get its data from the information system globally in use in the factory and must return its results for factory-floor execution [3].

4.1.3 Requirements

All Core requirements listed in Section 2 plus...

4.1.3.1 Outer Core Representational

1. abstraction - one example of the composition/decomposition concept of abstraction is the specification of a drilling operation. One could say at a high-level, “a part needs to be drilled”, and at a lower level say “Part A needs to be drilled by Drilling Machine B at 12 p.m. with a set-up task occurring before the operation and break-down task happening afterwards.” [6]
2. alternative tasks [55]
3. associated illustrations and drawings - a drawing of a part in a CAD package may be one example. [6]
4. complex groups of tasks [6]
5. complex resource characteristics [37][55]
6. complex sequencing of tasks [34][47][55]
7. complex task representation and characteristics - a complex characteristic of a gripper may be that it can perform two operations: transport and fixturing. [1][37]
8. concurrent tasks [55]
9. conditional tasks - a sanding task may only need to be performed if the surface roughness is greater than a certain value. [6]
10. confidence levels [6]
11. constraints [6]
12. date(s), time(s), and/or multiple duration(s) [6]
13. implicit/explicit resource association - a cutting tool may need a holder in order to work properly. One might never care exactly what holder was used, only that one existed and was used during the operation. [6]
14. iterative loops - in order to get a surface finish to a certain measurement, one may have to perform a task an unknown amount of times until that condition is met. [6]
15. manual vs. automated tasks [6]
16. manufacturing product quantity - also known as manufacturing order unit or batch size. [6]
17. material constraints [6]
18. parallel tasks [55]
19. parameters and variables [6]
20. pre- and post-conditions [6]
21. queues, stacks, and lists [6]
22. resource categorization and grouping [6]
23. resource location - this may be used for shop floor modeling and path optimization. [55]
24. resource/task combined characteristics [55]
25. serial tasks [55]
26. state existence constraints [6]
27. state representations - assembly tasks would be a strong candidate for state representation since the actual process of assembling the parts if not of primary importance. [6]

28. temporal constraints [6]
29. uncertainty/variability/tolerances [6]

4.1.3.2 Outer Core Functional

1. ability to insert/attach a highlight [6]
2. complex precedence of tasks [1][56]
3. deadline management [6]
4. dispatching [6]
5. eligible resources [6]
6. exception handling and recovery [6]
7. information exchange between tasks [6]
8. mathematical and logical operations [6]
9. support for task/process templates - some attributes that may need populating in a process template are speeds and feeds of machinery. [6]
10. support for simultaneously maintained associations of multiple levels of abstraction [6]
11. synchronization of multiple, parallel task sequences [6]

4.1.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, and constraints - one such rule may be "Only use machine X for one hour at a time because it is expensive to run." [6][49]
2. configuration management data [16]
3. forecasts and customer orders [6]
4. priority attributes [6]

Planning/Scheduling/Quality/Analysis

1. analysis characteristics [6]
2. co-existence of plans and resolution of conflicts - for example, a maintenance plan might need a resource to be cleaned the same time a process plan needs to use it to create a product. [6]
3. critical tasks [6]
4. predictive and time-dependent resource availability [6]
5. resource/system/process monitoring and feedback [7][55]
6. task/process performance measurement [6]
7. what-if analysis [6]

Real-Time/Dynamic

1. event signaling and notification [16]
2. process/resource status [6]
3. resource amount and availability [6][55]
4. resource behavior during processing time - an example may be a cutting tool's degradation during process time. [7]
5. resource/system/process monitoring and feedback [7][55]
6. track in-progress goods [6]

Process Intent

1. task/process purpose - the creation of a hole feature as shown in a CAD model may be the intent behind a drilling process. [6]

Aggregate Resource/Process

1. characteristics of a groups of resources - for example, if two machines share a work volume, only one machine can occupy that work volume at ant given time. [37][55]
2. implicit process association - for example, you may always need to perform a calibration task on a machine before one can perform a given task. [6]

Production Scheduling Application

1. production type data [73]
2. dynamic rescheduling [6]

4.2 Process Planning

4.2.1 Overview

A process plan specifies what raw material or components are needed to produce a product, and what processes and tasks are necessary to transform those raw materials into the final product [28]. Process planning is therefore the task of precisely specifying how to manufacture a particular product. As such, process planning forms the link between design and manufacturing.

Even in the smallest of companies, a need for a fully documented process plan exists. Repeatability of product quality, documented solutions to previously manufactured products, and a common process in which multiple machinists create a product are some of these advantages [55].

There are many subtasks involved in process planning for machined parts. They are, in sequential order: identify manufacturing features and part outline; divide up unwanted material to give discrete tasks; determine alternative product methods for each task; choose methods and sequence tasks to form basic plans; determine times, cutting conditions, etc. to give detailed plan.

In Computer Aided Process Planning (CAPP), there are two main approaches: variant and generative process planning. Variant CAPP systems are based on the retrieval of stored plans. These plans are then edited to accommodate any variations between the retrieved plan and the desired plan. The most important aspect of variant systems is their method of deciding the most appropriate plan to retrieve to be adapted for each new part. Group technology is a popular coding and retrieval technique for such a procedure. Generative CAPP systems generate plans from scratch for each individual component; they do not rely on stored plans [10]. Because of this, greater flexibility for product variation is possible and the potential for full automation exists. A generative system needs general knowledge about manufacturing processes as well as more specific information about available machines, tools, etc.

4.2.2 Current Practices

For the most part, process planning in industry is done manually. A person (the process planner) makes many of the decisions on how a product is to be manufactured using only his past experience and his basic knowledge as a guide. Process planners are typically highly experienced engineers, usually in short supply. Because of this shortage and the long amount of time it takes to train them, there is much interest in capturing process knowledge electronically. This would allow the knowledge to stay even when the process planner leaves. Hence, many companies are turning toward CAPP.

In addition, many companies are experiencing a trend toward smaller batches and higher product diversity. A major current concern is therefore for increased flexibility in manufacturing. This, of course, tends to make process planning costs more significant. and places an even greater load on the staff responsible. CAPP can help reduce cost and transfer the responsibility out of the staff’s hands and into the computer.

There are many potential benefits to capturing and representing process plans in a standard, computerized format. Some of them include:

1. process rationalization and standardization, leading to decreased cost and improved quality
2. increased productivity of planners, leading to reduced lead times
3. retention of expertise
4. integration with other applications

4.2.3 Requirements

All Core requirements listed in Section 2 plus...

4.2.3.1 Outer Core Representational

1. abstraction - there may be various representational levels of process plans, used for a variety of purposes. [51][56]
2. alternative task [37][55][56]
3. associated illustrations and drawings - for explanatory purposes. [11]
4. complex groups of tasks - there could be a group of tasks making up a sub-section of a process plan that may have a special significance (i.e. tasks that have a special importance in assuring the product functions properly). [1][30]
5. complex resource characteristics [34]
6. complex sequencing of tasks - to allow for non-linear process plans. [11]
7. complex task representation and characteristics [11]
8. concurrent tasks [11]
9. conditional tasks [11]
10. confidence levels [11]
11. constraints [37]
12. date(s) and time(s) and/or multiple duration(s) [11]
13. iterative loops - in order to get a surface finish to a certain measurement, one may have to perform a task an unknown amount of times until that condition is met. [11]
14. manufacturing product quantity - this may help to decide which type of operation is most cost effective. [11]
15. material constraints [37]
16. parallel tasks [11]
17. parameters and variables [11]
18. pre- and post-conditions [37]
19. resource categorization and grouping [51]
20. resource/task combined attributes - for characteristics of a resource that are dependent on the task its performing (i.e. non-preemption). [11]
21. serial tasks [11]
22. state existence constraints [37]
23. state representations [69]
24. temporal constraints [11]
25. uncertainty/variability/tolerances [11]

4.2.3.2 Outer Core Functional

1. ability to insert or attach a highlight - could be used to show important stages to someone in production. [11]
2. complex precedence of tasks [11]
3. convey the ancestry or tasks - for example, a drilling task may be a specialization of a machining task which may be the specialization of a job-shop task. In this sense,

characteristics of a job-shop task would always apply to a machining task and characteristics of a machining task would always apply to a drilling task.[30]

4. deadline management - to ensure the duration of the entire process plan does not exceed the allotted time. [11]
5. eligible resources - could included the types of resources that could be used for a given task. [1]
6. exception handling and recovery - a description of alternative methods of accomplishing a goal when an operation fails. [11]
7. information exchange between tasks - this would also allow information from one task to be available for subsequent tasks to allow for on-the-fly decision making. [11]
8. mathematical and logical operations [11]
9. support for task/process templates - to allow for population of specific attributes during the production phase. [11]
10. support for simultaneously maintained associations of multiple levels of abstraction - this would be helpful when a process planner is not sure exactly what resource to use but is able to describe the type of resource and what type of function it needs to perform. [51]
11. synchronization of multiple parallel tasks - when two or more otherwise dissimilar tasks have to happen at the same time[56]

4.2.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, constraints - this could play a factor in the decision making process of deciding what type of task should be used to accomplish a particular goal. [30][49][73]
2. configuration management data [11]
3. customer driven task/process specifications and constraints [30]
4. priority attributes [11]

Planning/Scheduling/Quality/Analysis

1. analysis characteristics - this would allow a process planner to evaluate a number of different alternatives to determine which would best accomplish the goals. [6]
2. critical tasks [11]
3. co-existence of plans and resolution of conflicts [11]
4. predictive and time-dependent resource availability - to allow a process planner to take an educated guess at what resources will be available at a given time. [11]
5. prescriptive task behavior - this could be derived from past data of process output.[11]

6. provide validation of the whole process - to ensure a process plan will occur as expected. [11]

Process Intent

1. intentional dimension of a task/process, or goals - this would allow a process planner to determine if a previously created plan can be used for an existing problem. [1][37]

Aggregate Resource/Process

1. implicit process association - this would included process to process dependencies and recommended practices. [11]

Process Planning Application

1. clamping surfaces [11]
2. features to be machined [30]
3. production type data [73]
4. datums and offsets [11]

4.3 Simulation

4.3.1 Overview

In general, the purpose of a simulation model is to allow observation about a particular system to be gathered as a function of time. From that standpoint, there are two distinct types of simulation: discrete and continuous. [66]

In discrete simulation, observations are gathered only at selected points in time when certain changes take place in the system. [66] Discrete event simulation is the computer-based execution of events to predict the performance of a system. For example, in the simulation of a manufacturing plant, the casting, machining, and assembling of parts are individual events which are executed in sequence to product a finished part [54]. A simulation system models the flow of materials from place to place and the transformation of that material at each processing step. It is designed to allow an engineering analyst to describe a material processing facility in terms of material movement and material transformation [54].

Continuous simulation requires that observations be collected continuously at every point in time.[66] This type of simulation is used when discrete points in time are not enough to completely simulate the process - a continuous re-enactment is necessary to determine all cause and effect relationships within the process.

Although continuous simulation has important applications, most attention has been directed to discrete simulation. The main reason is the wide range of problems that exist in this area. Additionally, the implementation of continuous simulation appears straightforward, whereas that of discrete simulation usually requires more originality and creativity on the part of the user. For that reason, the focus of this requirements study is on discrete simulation.[66]

For example, the development of a system model, commonly called a virtual plant, could allows the engineering analyst to assess material movement and inventory, waste generation, and personnel and equipment utilization within an integrated plant environment. By building a simulation model to encompass a large system, process chokepoints and complex interdependencies between subsystems can be identified, and proposed alternatives can be assessed. Since the development and prototyping of a process technology is an expensive proposition, a candidate task can first be inserted into a virtual plant simulation to gain insight into the system impacts. [54]

Tools developed for discrete event simulation are normally divided into two major categories: simulation languages and simulators. Discrete event simulation languages are general in nature and require programming to perform analysis, although some simulation programs do include built-in manufacturing and material-handling features. Simulators have built-in tools (examples are machines and conveyors) which allow non-programmers to rapidly construct models. [54]

4.3.2 Current Practices

For most simulation packages, the modeling of the process as well as the logic behind it is done within the simulation language itself. Additional work is usually needed to collect necessary input data for use within the model. Debugging is essential for ensuring that the logic of the model is correct. It usually involves tracing the simulation computations as a function of time and checking to see if the results of the model “make sense”. [66]

Simulation models are usually created for a specific purpose. Because of this, the representation of the process to be simulated is very biased toward that purpose. Attributes of the process that are important to the purpose are emphasized and modeled in detail while attributes that aren’t as important are incomplete or not modeled at all. Because of this, the model is only complete enough to be used for certain circumstances.

Existing simulation models generally represent at least three types of “nodes”: source nodes, queue nodes, and facility nodes. Source nodes represent where an item to be processed came from, queue nodes represent where the item waits if the facility is not yet ready to process it, and facility nodes represent the resource which will do the processing. Some simulation packages also have auxiliary nodes to provide additional modeling functionality. [66] There are a few other concept that are usually implicitly or explicitly captured. They are: the concept of time, flow control, system rules and characteristics. All three of these concepts are essential to ensure that the simulated process closely imitates the actual run. [66]

4.3.3 Requirements

All Core requirements listed in Section 2 plus...

4.3.3.1 Outer Core Representational

1. abstraction - subsystem simulations can be linked and abstracted enabling accelerated analysis of a process without being burdened by the simulations of lower levels in the hierarchy. [59]
2. alternative tasks [59]
3. complex groups of tasks - a group of tasks that all occur on a specific machine would allow one to simulate only that machine to determine load and performance. [66]
4. complex resource characteristics [33][59]
5. complex sequences of events [33][66]
6. complex task characteristics [59][66]
7. concurrent tasks [66]
8. conditional tasks [66]
9. confidence levels - important to determine the probability of an event occurring as expected. [66]
10. constraints [66]
11. date(s), time(s), and/or multiple duration(s) - for analysis purposes [66]
12. implicit/explicit resource association - to ensure ALL necessary resources are available. [66]
13. iterative loops [66]
14. manual vs. automated tasks [66]
15. manufacturing product quantity [66]
16. parameters and variables - could be used to create a histogram. [66]
17. pre- and post-conditions [66]
18. queues, stacks, and lists - queues are of particular importance to simulation in order to represent where a product is stored while it waits to be processed. [66]
19. parallel tasks [66]
20. resource categorization and grouping - could be use to group eligible resources for a process. [66]
21. resource location - this attribute can be used to represent where a product originated and where it is to be processed. [66]
22. resource/task combined characteristics - characteristics such as non-preemption are important to model to ensure the simulation package behaves as it should. [66]
23. serial tasks [66]

24. state existence constraints [66]
25. state representations - to simulate processes where the state of the product at any given time is important. [59]
26. temporal constraints [66]
27. uncertainty/variability/tolerances [66]

4.3.3.2 Outer Core Functional

1. complex precedence [66]
2. deadline management - could be used for analysis purposes to determine if a process will finish in the allotted time. [59][66]
3. dispatching - described the rate at which products can be released from the source into the queue. [66]
4. eligible resources [66]
5. exception handling and recovery - to let the simulation package know what to do if the output of any given task is not what's expected. [66]
6. information exchange between tasks [66]
7. mathematical and logical operations - possibly used to perform analysis on the results of the simulation or to check conditions to see if a branch should be taken (such as IF-THEN-ELSE-ENDIF statements). [66]
8. synchronization of multiple, parallel tasks [33]

4.3.3.3 Extension Requirements

Business/Administrative

1. manufacturing rules - may play an integral role in the simulation package's decision making process. [33]
2. priority attributes [33][59]

Planning/Scheduling/Quality/Analysis

1. analysis characteristics - to interpret the results of the simulation. Some types of analysis include resource utilization, defects per unit cumulative for a line, defects per unit per workcell, resource cycle time, production bottlenecks, various types of costs, and runtime of simulation. [33][59][66]
2. critical task [33]
3. co-existence of plans and resolution of conflicts [59]
4. resource/system/process monitoring and feedback [66]
5. task/process performance measurements - this is of particular importance to simulation as it may be an output from a simulation package and possibly an input for analysis. [66]

6. task/process performance measurements [37][55]
7. tracking of changes in the system [66]
8. what-if analysis [59]

Real Time/Dynamic

1. event signalling and notification [66]
2. resource amount and availability [66]
3. resource/system/process monitoring and feedback [37][55]
4. tracking of changes in the system - to ensure the simulation model is constantly correct. [66]

Aggregate Resource/Process

1. implicit process association - this will let the simulation package know that when a given process is simulated, another process which is associated with it must be performed and simulated also. [66]

Stochastic/Statistic

1. account for randomness - can be modeled in terms of probability in an attempt to quantify previously captured data. [66]
2. descriptive manufacturing/performance variability - to be represented by a simulation package to quantify the possible deviation of a process from the norm. [60]
3. probabilities of down-time - can be depicted by a simulation package to show the probability of a resource being unavailable when needed. [66]
4. stochastic functionality [66]
5. stochastic properties [66]
6. uncertainty of sequences - when multiple sequences can be selected for a given task, a simulation package can show the chances of a particular one being chosen. [66]

4.3.3.4 Simulation Application

1. queue entry and exit rates - this is of particular importance in simulation because of its ability to tell which machines in the process is causing a bottleneck. If the entry rates for a machine's queue is large and the exit rate is small, the machine is creating a bottleneck. [66]

4.4 Work Flow

4.4.1 Overview

While the term "workflow" may be used to describe business process tasks at a conceptual level necessary for understanding, evaluating, and redesigning the business process, Workflow and Workflow Management (WFM) typically focus on business processes that are implemented as information processes. [27] In this sense, workflow specifies the information process tasks at a level that describes the process requirements for information system functionality. Workflow essentially captures methods, tasks, and the related software applications in a computer-interpretable form for modeling, simulation, and execution. In [43], Workflow is defined as "a tool set for the pro-active analysis, compression, and automation of information-based tasks and activities".

The original concept of Workflow was applied to the use of computer software to automate paper-driven processes and it focused on repetitive, predictable processes such as loan applications or insurance claims. While most Workflow software applications are developed for this transaction-based environment, workflow technology is being applied to other development environments including:

- ad hoc - process rules are not easily defined
- object-oriented - rules for the pieces are known, but the process is reconstructed in different ways
- knowledge-based - relationship rules are known, but process rules require heuristics

Workflow is useful to an organization because understanding and controlling of both information and document flow can:

- improve the integrity and reliability of information
- improve productivity through the identification and elimination of bottlenecks
- increase communication and quality
- enable change management through the implementation of process metrics

4.4.2 Current Practices

Workflow, now typically considered a distinct technology, has evolved from a variety of origins, including image management, document management, electronic mail, computer aided software engineering, and pure workflow modeling. All workflow systems, though, contain generic components which interact in a variety of ways. Users of workflow systems could achieve significant gains with a standard process representation that could be used among different tools. This ability becomes even more valuable as an enterprise expands to include several remote companies using Electronic Document Interchange to process information.

There is an identified need for standards in workflow technology, specifically in terminology, interface metaphors, methods, application, and integration techniques. The Workflow Management Coalition (WfMC), established in 1993, is working to develop API standards, terminology standards, interface standards, and interoperability capability. Information from the WfMC's glossary of terms and other available reports was used in this requirements document. [71]

4.4.3 Requirements

All Core requirements listed in Section 2 plus...

4.4.3.1 Outer Core Representational

1. abstraction, or task nesting - to enable different views of the workflow. For example, management would be interested in the higher levels of the business process, while lower levels of abstraction capture the specifics to implement the workflow. [27]
2. complex grouping of tasks - to enable assigning a resource(s) to an entire group of tasks [27]
3. complex resource characteristics (e.g., mapping resources to (often multiple) capabilities - to facilitate dynamic load balancing) [27]
4. complex sequences of tasks to represent task structure (control flow) (e.g., alternative tasks, parallel tasks [27] and serial, parallel, synchronized activities that converge into (AND-Join) or split from (AND-Split) one thread of control or two or more non-synchronized activities join into (OR-Join) or split from (OR-Split) a single activity [71]
5. complex task representation and parameters - different types of tasks (manual vs. automated) can affect execution behavior [63]
6. conditional tasks, where transition conditions must be met for continuing from current activity to next activity(s) [71] [43]
7. constraints on task or resource [27]
8. date(s), time(s) and/or multiple duration(s) [27]
9. iterations, or workflow loops, involving repetitive execution of activities until a condition is met [71]
10. manual vs. automated tasks [27]
11. product quantities - the number of work items to be produced
12. parameters and variables - important for implementation support
13. queues, stacks, lists [71]
14. complex resource categorization or grouping (e.g., organizational group capabilities - attributes or skills that can be assigned to a resource for the purpose of achieving organizational objectives) [71]
15. resource location - useful in analysis to improve workflow efficiency [27]
16. resource/task combined characteristics

17. state representations - activity instances with the following functions: start, resume, suspend, terminate, view (e.g., status) and the following states: active, inactive, completed [71] for transaction coordination - associated with recovery, to maintain the state of each task with respect to a specific workflow [27]
18. uncertainty/variability/tolerance

4.4.3.2 Outer Core Functional

1. complex precedence/intertask dependencies (e.g., order dependency - if both A and B occur, then A precedes B, existence dependency - if A sometimes occurs, then B sometimes occurs) [63]
2. deadline management (e.g. for document management) [43]
3. dispatching [43]
4. eligible resources - resources that have the capability of performing a given task. [43]
5. exception handling - to specify actions if a workflow is not completed or a task fails [27] and recovery - when a workflow is aborted midstream - how to undo completed or partially completed tasks [27]
6. information exchange between tasks (data flow) (e.g. tasks that require data from other tasks) [27]
7. mathematical and logical operations [32]
8. support for task/process templates
9. synchronization of multiple, parallel task sequences

4.4.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, constraints - internal and external rules or constraints on entities [27]
2. configuration management - for document management [27]
3. customer driven process specifications and constraints
4. priority attributes - based on date received, date created, or user-defined [27]

Planning/Scheduling/Analysis/Quality

1. critical tasks - useful in analysis of workflow performance, identification of bottlenecks [27]
2. queue times - for analysis [71]
3. workload attributes - throughput, volume [27]
4. workflow monitoring to determine bottlenecks, completion times, workload [27]

Real-Time/Dynamic

1. event signaling and notification (e.g., one task notices events of another task and takes action, deadline management) [27]
2. dynamic workflow modification - including changing of workflow variables such as volumes, headcount (resources), task time, roles; changing task sequencing; adding new tasks into an executing workflow [63]

4.4.3.4 Workflow Application

1. invoked tool capability - enables automated applications to be invoked by the workflow management system [27]
2. manual process activity and definition (that cannot be automated using a workflow management system) and workflow process activity (that can be automated) [71]

4.5 Enterprise Engineering and Business Process Re-engineering

4.5.1 Overview

Business Process Re-engineering (BPR), a very hot topic in the manufacturing industry today, can be defined as "the fundamental re-thinking and redesign of business processes to achieve dramatic improvements in critical contemporary measures of performance such as cost, quality, service, and speed." [31] The Society for Enterprise Engineering (formerly the IDEF Users Group) defines Enterprise Engineering (EE) as "...that body of knowledge, principles, and disciplines having to do with analysis, design, implementation and operation of an enterprise. Enterprise Engineering views the enterprise as a system of cultural, process, and technology components that interact to accomplish strategic objectives and goals. The EE process addresses the critical aspects of how to design and improve all elements associated with the total enterprise through the use of engineering and analysis methods and tools to more effectively achieve its goals and objectives."

4.5.2 Current Practices

Many methods have been employed to implement BPR or EE, including techniques and methods such as total quality management and continuous process improvements. While these methods have had some successes, there is a growing understanding that complex business processes can only be understood, analyzed, and modified using analytical engineering methods, including automated modeling and simulation tools.

These tools enable users to visualize bottlenecks and communicate changes or “sell” proposed process improvements. The documentation of processes in a common format allows for easier modification of existing processes to represent the new processes. Tools with discrete event modeling provide a way to vary the timing of events, tasks, and processes for “what if” analyses. [67]

Even with the use of these tools, implementations of BPR in manufacturing companies have been met with varying degrees of success for a variety of reasons. Part of the reason is that most traditional modeling and analysis tools are simplistic. [32] The typical static process modeling tools express what occurs within a process and the process inputs and outputs — essentially the flow of work products from one work unit to another. While this information is necessary, business planners also need dynamic, behavioral modeling to reveal how a business process operates, the conditions under which it operates, and the events that take place within it. This information is essential for answering the “why?” and “what-if?” questions that are central to re-engineering. Additionally, models need to effectively and adequately communicate why process changes are needed or how the proposed process change will work.

To be useful to business process re-engineering, a process model must be able to represent business operations in the context of strategic goals, company culture, and in terms of business operation efficiency measures. [40] Jarzabek and Ling state that BPR and simulation tools must include a precise internal information model that comprehensively covers many aspects of business structure and dynamics, and that records many kinds of dependencies between the business entities that have to do with the understanding of business operations. There needs to be multiple views of the model, including a functional view — what is happening in the process and what it produces — and a behavioral view — how, why, and when the process happens.

It is generally agreed that the practice of business re-engineering is still more of an art than a science and that a more systematic approach to the design of business processes, i.e., the development of models that provide appropriate representations of the knowledge that is needed for understanding and reasoning about business processes, is needed. [72] Relevant aspects to be included in this model are an enterprises’ processes, strategies, organizational structure, resources, goals, constraints, and environment. [25]

4.5.3 Requirements

All Core requirements listed in Section 2 plus...

4.5.3.1 Outer Core Representational

1. abstraction - to view process at multiple views or levels, i.e., enterprise, business, and operations levels [70], for hierarchical decomposition [32]
2. complex grouping or categorization of tasks and / or resources - [32] states this as a need for repositories, or a collection of similar or different objects
3. complex resource characteristics - resources can provide multiple functions [72]

4. complex sequences / task structure - control flow (e.g., parallel tasks, conditional tasks, alternative tasks) [32]
5. complex task representations - to enable “transactions,” non-value added tasks which may move product(s) from one operation to the next without changing the product [32]
6. constraints [41]
7. date(s), time(s), and/or multiple durations (i.e., estimated, average) associated with tasks [41]
8. iterative loops [41]
9. manual vs. automated tasks [32]
10. parameters and variables [41]
11. queues, stacks and lists - a special, ordered type of repository [32]
12. resource location[41]
13. resource/task combined characteristics - alternate or multiple resources (e.g. people) for same task, with varying attributes (e.g. activity time) for the same activities [41]
14. resource/task combined characteristics - resource pre-emption [41]
15. state representation [41]
16. uncertainty/variability/tolerance - variable attributes to represent uncertainty (e.g., task duration with some tolerances) [41]

4.5.3.2 Outer Core Functional

1. ability to insert or attach highlights (milestones) [41]
2. complex precedence - on operations where multiple inputs that must all occur before an operation can begin (i.e., A and B and C must be present before an operation can begin) [32]
3. support dispatch rules (e.g., first-in-first-out, last-in-last-out, priority) [32]
4. eligible resources [41]
5. information exchange between tasks [32] [41]
6. mathematical and logical operations (e.g., if-then logic to model complex situations such as conditions to model decision checks) [32]
7. support simultaneously maintained associations of multiple levels of abstraction [70]
8. synchronization of multiple, parallel task sequences [41]

4.5.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, constraints - internal and external rules or constraints on entities (e.g., safety regulations as a characteristic of a work entity or a business policy of no more than 10 employees per manager)
2. configuration management - esp. process logging
3. customer driven process specifications and constraints
4. priority attributes

Planning/Scheduling/Analysis/Quality

1. analysis/performance characteristics such as resource utilization, throughput, cost, cycle time, likelihood of error, quality of service to customers
2. critical tasks
3. task/process performance measurement
4. dynamic model modification -for "what-if" analysis
5. optimization
6. what-if analysis

Process Intent

1. access to past and present decision rationales
2. intentional dimension of processes, or goals (e.g., activities have goal attribute) Some of these goals may be vague and not clearly defined.
3. task/process purpose
4. relationship between task and goal and resource and goal
5. value-added attributes - e.g.,value of business process to customer - could also be considered a "relationship identifier" requirement - e.g., AddsValueTo (Process, Customer). The value added can have different degrees of strength and be used to identify critical tasks.

Stochastic/Statistics

1. stochastic capabilities for resources, tasks, durations, and processes

4.5.3.4 Enterprise Engineering and Business Process Reengineering Application

1. conceptual entities, such as ideas or knowledge (e.g., an engineer's knowledge of the result of a test) (this could be a specific case of abstraction, incompleteness, ambiguity, accurate - not precise)

4.6 Project Management

4.6.1 Overview

In A Guide to the Project Management Body of Knowledge (PMBOK) [19], Project Management is defined as, "the application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations from a project. Meeting or exceeding stakeholder needs and expectations invariably involves balancing competing demands among:

- scope, time, cost, and quality
- stakeholders with differing needs and expectations
- identified requirements (needs) and unidentified requirements (expectations)."

Essentially, Project Management is concerned with five basic issues[61]:

- outputs (What is being produced?)
- process steps (How will it get done?)
- resources (What is needed?)
- schedule (When is it needed?)
- status (How is it going?)

Project Management tools are used for project planning, management, control, and reporting. During the planning phase, PMs want to be able to determine things like earliest and latest start and finish dates for activities and projects, the critical path for the project, resource constraints. Project Management for control enables PMs to track progress against the plan and to make adjustments as required to (hopefully) keep things under budget and within schedule. Overall, project plans are used to guide project execution, document planning assumptions and planning decisions regarding alternatives chosen, facilitate communication, define key reviews and milestones, and provide a baseline for progress measurement and project control.

4.6.2 Current Practices

The major complex requirements for modeling project processes are for resource loading, risk analysis, and detailed performance tracking. While nearly all project management tools are for project control, there is a clear need for risk analysis that can support the exploratory description of a project in the early stages when there is much uncertainty. PERT (Program Evaluation and Review Technique) does this to some extent (and most project management software is derived from PERT and/or Critical Path Method (CPM)), allowing a planner to define activities, sequences, and three estimates of the resources required for each activity (optimistic, most likely, and pessimistic). Many practitioners believe that there is a need for tools to enable the development of deterministic plans for control and probabilistic models for forecasting. (It is interesting to note that others seem to question the reality of forecasts based on stochastic models. The real problem appears to be a lack of data with which to model “similar tasks”. Tasks often have little similarity from project to project, even within projects. Essentially, no amount of stochastic modeling can create information that was not there to begin with.)

Current project management systems are based on PERT and/or CPM techniques and use proprietary data formats, prohibiting the transfer of project plans from one project management system to another. This is a real problem when multiple companies are participating on one project or when a various sub-contractors work for a single prime as was discovered by the U.S. Army Corps of Engineers. To address this, the U.S. Army Construction Engineering Research Laboratories (USACERL) develop the Standard Data Exchange Format (SDEF) to enable CPM software systems to exchange data. [20] As of 1990, five vendors had demonstrated compatibility with SDEF. Contractors for Corps of Engineers projects requiring electronic schedule information are required to use this format. Information from the SDEF specification was used in this requirements document.

4.6.3 Requirements

All Core requirements plus...

4.6.3.1 Outer Core Representational

1. abstraction (composition/decomposition) - the ability to break tasks into smaller tasks, to link subprojects to master project [48] [19]
2. complex grouping of tasks - to assign resources (costs) to “work packages” [19]
3. complex resource characteristics [19]
4. complex sequences (e.g., alternative tasks, concurrent tasks) [19]
5. complex task representation and parameters (e.g., “hammock” tasks whose times are dependent on start date of preceding task and finish date from succeeding task) [20]
6. confidence levels - esp. to express level of confidence in duration estimates for project planning and risk analysis [19]
7. constraints - ladder activities e.g., temporal constraints on the beginning of an action in relationship to the beginning of another. Also to define precedence relationships, such as Finish-to-Start (FS), Start-to-Start (SS), Finish-to-Finish (FF), and Start-to-Finish (SF) [20]
8. date(s) and time(s) and/or multiple duration(s) - earliest, latest start and finish times [20]
9. implicit/explicit resource association
10. resource categorization and grouping - resource pools [19]
11. resource/tasks combined characteristics - e.g., different resources may take different times to perform the same task [19]

12. state representations - to describe when tasks are complete, partially complete, etc.[19]
13. uncertainty in durations, costs, and variability in time to complete - to account for learning curve [19]

4.6.3.2 Outer Core Functional

1. ability to insert or attach a highlight - milestones (events at a point in time) [61]
2. activity precedence - any activity can be preceded by or directly followed by one or more activities [20]
3. deadline management [19]
4. eligible resources [19]
5. mathematical and logical operations [32]
6. support for tasks/process templates [48]
7. support for simultaneously maintained associations of multiple levels of abstraction
8. synchronization of multiple, parallel task sequences [19]

4.6.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, constraints - e.g., working and non-working days, hours [48]
2. configuration management of information and processes - to handle multiple time zones [19]
3. task prioritizing - to resolve conflicts for demand for same resource [19]

Planning/Scheduling/Quality/Analysis

1. actual time spent - for control, to be used with forecasting capabilities. Allows managers to forecast time to be spent and then collect actual hours spent for timesheet recording [19]
2. analysis characteristics to enable critical path calculation, risk analysis, and determination of lag time (time delay between connected activities) and float time (amount of flexibility in starting and completing tasks), driving and non-driving resources (to enable accurate, leveled schedules), conflicting demands for the same resource [19]
3. co-existence of plans and resolution of conflicts when two or more project plans use the same resource at the same time [19]
4. status calculations: Budgeted Cost for Work Scheduled (BCWS), Budgeted Cost for Work Performed (BCWP), Actual Cost for Work Performed (ACWP) — and resulting variance analysis [19]

5. time-dependent resource availability - to account for coming and going of resources [19]

Real Time/Dynamic

1. resource interruptions - e.g., a resource becomes ill when a task in 50% complete, returns and completes it, with a delay [19]
2. resource amount and availability [61]
3. resource/system/process monitoring and feedback [61]
4. track changes in the project plan [61]

Aggregate Resources/Processes

1. parallelism - a resource can perform multiple, simultaneous functions at one time [61]

Stochastics / Statistics

1. uncertainty in sequences of activities - for risk analysis [19]

4.6.3.4 Project Management Application

1. Work Breakdown Structure element - consisting of small units of work [61]

4.7 Product Realization Process Modeling

4.7.1 Overview

Manufacturing firms use various methods to develop process models for understanding, improving, and even managing the activities necessary to bring a product from early concept through fabrication and assembly. In [46], the term *product realization process (PRP) modeling* is used to distinguish this activity from the more generic term *process modeling*. Here, a PRP model is defined as "...computer-interpretable description of the human and machine activities and their interactions required to realize a... product. This may include early concept and configuration design activities, detailed design, prototyping, testing, fabrication, assembly and the many other activities within the scope of the realization process."

Like BPR and WFM, PRP focuses on the processes associated with completing a task. PRP specifically addresses the multi-functional complexities associated with designing and building products and the downstream implications of complex design process interactions. This can involve decision-making about the design as well as resource allocation and scheduling for engineering activities and coordination with marketing, finance, purchasing, and other functional groups within an enterprise. Early design decisions can be made more effectively with information about downstream life-cycle cost/schedule impacts. [45] To summarize, benefits include documentation of best practices, identification of bottlenecks and task redundancy, "what if" analyses of design alternatives, risk assessment for schedule and cost, archiving PRP processes, and training.

4.7.2 Current Practices

PRP models can be used for many applications such as documenting and revising engineering change order processes, investigating requirements for supplier chain integration, process modifications resulting from CAD integration efforts, and business process re-engineering efforts. Most companies that attempt PRP modeling use non-computer interpretable, static models that serve as guides for new process development, but are limited in their ability to be updated for future analysis and use. Industrial practitioners typically develop text-based models, perhaps with very basic flowcharting for a graphical depiction of processes. A company-proprietary guide may be used for ad hoc diagrammatic and textual descriptions. More sophisticated approaches may use PERT-based (Program Evaluation and Review Technique) models or IDEF-based (Integrated Definition Methodology) models. While a more in-depth analysis of these and other approaches with respect to representing process will be documented in a subsequent paper, these models are unable to address many of the complexities of multi-functional, new product development processes. For example, the time and cost uncertainty resulting from activity iterations and the presence of contingency activities are requirements not addressed adequately by these methods.

There is a growing need to share information and knowledge between process modelers and other engineering and business applications, yet no common standard or method has been developed to adequately address this. The corporate knowledge, along with the time invested in modeling a company's processes, can be locked up in closed applications, rendering companies incapable of migrating or sharing information with other process modelers and manufacturing applications. Moreover, competition has forced a push to more concurrent engineering that demands the integration and sharing of process data. While some vendors have developed conversion algorithms for simulation and IDEF modeling packages, there has been little research in this area.

4.7.3 Requirements

All Core requirements plus...

4.7.3.1 Outer Core Representational

1. abstraction - to represent processes during different stages of the product realization process, which may have different levels of confidence or different representational constructs or be undefinable in the early phases of product definition [45]
2. complex grouping or categorization of tasks and / or resources [46]
3. complex resource characteristics - resources (esp. people) can provide multiple functions [46]
4. complex sequences / task structure - control flow (e.g., parallel tasks, conditional tasks, alternative tasks, contingency tasks) [46]
5. complex task representations and parameters - to enable representation of non value-added tasks [46]
6. confidence levels - the degree of certainty that some attribute is true [46]
7. constraints - temporal, conditional, existence constraints on tasks, resources [46]
8. date(s). time(s), and/or multiple duration(s) (i.e., estimated, average) associated with tasks
9. iterations - design iterations required as design is modified, and a modified set of processes must be accomplished in each iteration [46]
10. manual vs. automated tasks
11. parameters and variables [46]
12. queues, stacks and lists
13. resource categorization and grouping [46]
14. resource location [46]
15. resource/task combined characteristics - a resource cannot be pre-empted when performing a specific task [46]
16. state representation
17. uncertainty/variability/tolerance - variable attributes to represent uncertainty (e.g., task duration with some tolerances) [46]

4.7.3.2 Outer Core Functional

1. ability to insert or attach highlights (milestones)
2. complex precedence - e.g., tasks where multiple inputs that must all occur before another task can begin [46]
3. deadline management
4. support dispatch rules (e.g., first-in-first-out, last-in-last-out, priority)
5. eligible resources [46]
6. exception handling and recovery [46]
7. information exchange between tasks - time dependent [46]
8. mathematical and logical operations (e.g., if-then logic to model complex situations such as conditions to model decision checks)
9. support task/process templates

10. support simultaneously maintained associations of multiple levels of abstraction [46]
11. synchronization of multiple, parallel task sequences [46]

4.7.3.3 Extension Requirements

Administrative/Business

1. business practices, rules, constraints - internal and external rules or constraints on entities
2. configuration management - esp. process logging

5.0 Summary

The goal of the first phase of this project was to create and categorize a comprehensive list of requirements that are necessary for representing process. The premise was that there are a large number of the requirements for representing process that are common to most applications that use process information, with very few requirements being application specific. If this is indeed the case, a common process representation should be possible. The common process representation, ultimately a standard process specification language, would be useful for improved integration of process-centric applications as well as provide the basis for powerful, robust applications.

There are a number of ways that a common process representation could be applied to improve integration. The first is enabling otherwise dissimilar applications to be interoperable by allowing them to "speak the same language." This could be done by providing internal converters to output the process specification language as a neutral file exchange format or by simply using the specification language as a software application's native format. This would eliminate the need for external conversion from one application's native representation to another's, a common and time consuming practice in industry today. The second is by allowing various applications that use process information to migrate toward a shared, common database that is based on the process specification language, thus eliminating the possibility of redundant and duplicate data. This information source must be comprehensive enough to include all of the individual requirements necessary for the various applications that use it.

Additionally, a process specification language that successfully addresses a comprehensive set of requirements will likely enable more robust applications than what currently exist today. This is important especially when considering that with increased automation and integration of all aspects of manufacturing, there is a growing need to better understand processes and to have a standard, robust method for process representation.

5.1 Observations

There was significant overlap in requirements for representing process among the applications under study for this phase of the project, supporting the supposition that a common process specification language is feasible. In fact, there are few requirements that are unique to any individual application (the "Application" requirements).

While the "Extension" and "Application" requirements tend to be application-specific, the "Core" and "Outer Core" requirements are common to virtually all of the business and engineering applications under study. Although the focus of this project was limited to design/manufacturing life-cycle applications, all of the core requirements and most of the outer core requirements appear to have general applicability to applications far beyond the manufacturing domain.

5.2 Future Work

With an understanding of the requirements necessary for representing process, the next step will be to analyze multiple existing representations to determine how well they each represent the process requirements. A matrix will be created to crosscheck the degree to which the various representations capture the various requirements. This work should provide an objective basis for identifying which representation or combination of representations provides the best coverage of the process requirements.

It will be the aspects of these various representations that best capture the process requirements that will be used as a basis for the next phase of the project, the synthesis of a process specification schema. It is therefore our hope that the concepts used in existing representations can be built upon thereby minimizing the need to invent new concepts.

To keep the scope of the project feasible, not all of the requirements gathered in the initial phase of the project will be modeled in the schema. However, enough requirements will be included for a prototype implementation, to be identified at the beginning of the next phase of the project. These requirements will include all of the core, most of the outer core, and the extension and application requirements that are applicable to the implementation. Because it is not the intent of the project to duplicate efforts that already have well-defined solutions, requirements that are represented in other models, such as a resource model or a product model, will not be included. Instead, a mechanism will be put in place in the process specification language to access this information from the other models.

Once the schema is created, a language syntax and grammar and one or more notations will be defined. After this is completed, one or more prototype implementations will be performed to ensure the completeness and ease-of-use of the process specification language. At this point, a validated candidate standard will be submitted to an appropriate standards body.

5.3 Concluding Remarks

At the conclusion of each phase of the project, a document will be produced that will include the relevant findings. Although each of these documents will represent a building block toward the ultimate goal of a standard, unified process specification language, each should also be able to stand alone as a single, identifiable topic area relating to process representation and specification.

It is the hope of the authors that this document will be used as a reference for future researchers and product developers as a comprehensive, definitive source for process representation requirements.

Bibliography

[1] Akira, Okano, “Computer-Aided Assembly Process Planning with Resource Assignment,” IEEE, Vol 93, 1993.

[2] Al-Ashaab, Ahmed H and Young, Robert I. M., “A Manufacturing Model: Supporting Concurrency in Injection Moulded Products Design,” in Proceedings of the ASME/ESDA Conference, 1994.

[3] Arnett, Matthew Flint, et al., Inside TCP/IP, Second Edition, New Riders Publishing, ISBN 1-56205-450-3, 1995.

[4] Baptiste, Philippe and Le Pape, Claude, “Disjunctive Constraints for Manufacturing Scheduling: Principles and Extensions,” in Proceedings of the Third International Conference on Computer Integrated Manufacturing, Singapore, 1995.

[5] Bay Networks Open System Interconnect (OSI), http://bbs-koi.uniinc.msk.ru/product/bay/routers/protocol/bridge/osi.htm, July 25, 1996.

[6] Bodington, C. Edward, Planning, Scheduling, and Control Integration in the Process Industries, McGraw-Hill Inc., 1995.

[7] Boning, Duane S., McIlrath, Michael B., “Conceptual Graphs and Manufacturing Processes,” Massachusetts Institute of Technology Internal Report, 1994.

[8] Boning, Duane, McIlrath, Michael, “Semiconductor Process Representation Information Model Overview,” http://www-mtl.mit.edu/CFI/SPR/wmwork/www/imov_1.html, Cambridge, MA, April 20, 1995.

[9] Catron, Bryan A., Ray, Steven R., “ALPS: A Language for Process Specification,” *International Journal for Computer Integrated Manufacturing* **4**(2), 1991.

[10] Chang, T.C., Expert Process Planning for Manufacturing, Addison Wesley, 1990.

[11] Chang, Tien-Chien, Wysk, Richard A., An Introduction to Automated Process Planning Systems, Prentice Hall, 1985.

[12] Coen, G., personal conversation regarding on-going research. Boeing Helicopters, May 30, 1996.

[13] Computer Science and Telecommunications Board and Manufacturing Studies Board, National Research Council, “Information Technology for Manufacturing: A Research Agenda”, National Academy Press, Washington, D.C., 1995.

[14] “Concurrent Engineering Fundamentals: Integrated Product and Process Organization”, Volume 1, Biren Prasad, Prentice Hall PTR, 1996.

[15] Cook, Nick, “Integrated Modelling of Products and Processes Using Advanced Computer Technologies,” ESPRIT, http://www.twente.research.ec.org/esp-syn/text/2165.html, July 18, 1996.

[16] CS/Capp II Version 6.5 System Manager’s Manual, CIMx, May 1994.

[17] Davies, Byron, “A Computational Framework for Manufacturing Process Specification,” in Proceedings of the DARPA/SRC CIM-IC Workshop, Stanford, CA, August 1988.

[18] Davies, Byron, “Process Specification and Process Knowledge in Semiconductor Manufacturing,” Stanford University ICL No. 91-012, Stanford, CA, March 1991.

[19] Duncan, W. R., A Guide to the Project Management Body of Knowledge, PMI Standards Committee, Program Management Institute, 1996.

[20] East, E. W., “The Standard Data Exchange Format for Critical Path Method Scheduling,” US Army Corps of Engineers Construction Engineering Research Laboratories (USACERL) Technical Report-95/40, September 1995.

[21] Eirich, P., “Enterprise Modeling: Issues, Problems and Approaches,” Draft Report of the ICEIMT Special Interest Group, International Conference on Enterprise Integration Modeling Technology, June 1992.

[22] Electronic mail communication to FSTAB members from Gene Meieran, May 28, 1996.

[23] Eng, Lawrence, et al., Computer Integated Manufacturing (CIM) Application Framework Specification 1.3, SEMATECH Technology Transfer #93061697F-ENG, 1996.

[24] ESPRIT 6805 COMPLAN, Concurrent Manufacturing Planning and Shop Control for Small Batch Production, http://www.mech.kuleuven.ac.be/pma/project/complan/complan.html, July 5, 1996.

[25] Fraser, J., “Managing Change through Enterprise Models,” in Proceedings, Applications and Innovations in Expert Systems II, SGES Publications, 1994.

[26] Genesereth, M., Fikes, R., “Knowledge Interchange Format Version 3.0 Reference Manual,” http://logic.stanford.edu/kif/Hypertext/kif-manual.html, Stanford, CA, May 15, 1996.

[27] Georgakopoulos, D., M. Hornick, and A. Sheth, “An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure,” Distributed and Parallel Databases **3**(2): 119-154, 1995.

[28] Groover, M.P., Automation, Production Systems and Computer Integrated Manufacturing, Prentice Hall, 1987.

[29] Halsall, DN, Muhlemann, AP, Price, DHR, "A Production Planning and Scheduling Framework for Smaller Manufacturing Enterprises," Emm Lane, Bradford, West Yorkshire, 1995.

[30] Halevi, G., Weill, R.D., Principles of Process Planning, A Logical Approach, Chapman & Hall, England, 1995.

[31] Hammer, M., and Champy, J., Reengineering the Corporation: A Manifesto for Business Revolution, Harper Collins, 1993.

[32] Hansen, G. A., "Tools for Business-Process Reengineering," IIIE Software **11**(5), Sept. 1994.

[33] Hurrion, R.D., Simulation: Applications in Manufacturing, Springer-Verlag, Bedford, UK, 1986.

[34] "IE 562: EXPERT SYSTEMS SEMESTER PROJECT," http://cac.psu.edu/~tool100/, July 5, 1996.

[35] "ILOG SCHEDULE/ILOG SOLVER White Paper," http://www.ilog.fr/ilog/Products/Schedule/wp.html, July 5, 1996.

[36] ISO, Product data representation and exchange: Part 1: Parts Library - Part 1: Overview and Fundamental Principles, ISO Standard 10303-1, 1995.

[37] ISO, Product data representation and exchange: Part 49: Integrated generic resources: Process structure and properties, ISO Standard 10303-31, 1995.

[38] ISO , Product data representation and exchange : Part 213 : Application Protocol: Numerical Control process plans for machined parts, ISO Standard 10303-213, 1995.

[39] "Information Technology for Manufacturing: A Research Agenda," Committee to Study Information Technology and Manufacturing, National Academy of Science, 1995.

[40] Jarzabek, S., Tok Wang Ling, "Model-based Design of Tools for Business Understanding and Re-engineering," in Proceedings. Seventh Annual Workshop on Computer-Aided Software Engineering (Cat. No. 95CB35827), 1995.

[41] Johansson, H. J., McHugh, P., Pendlebury, A. J., Wheeler III, W. A., Business Process Reengineering: Breakpoint Strategies for Market Dominence, John Wiley and Sons, 1993.

[42] Kemmerer, Sharon (editor), "Initial Manufacturing Exchange Specification (IMES): IMES Concept Document for Manufacturing System Integration ," (title may be changed) *to be published as a NIST Internal Report in late 1996*.

[43] Koulopoulos, T. M., The Workflow Imperative: Building Real World Business Solutions, Van Nostrand Reinhold, 1995.

[44] Lee, Seongkyu, et al., "A Process Plan Representation Model for Shop Floor Control", DRAFT, 1996.

[45] Lyons, K.W., Duffey, M.R., Requirements, Methods, and Research Issues for Modeling the Product Realization Process, IFIP Working Conference "Re-engineering the Enterprise", Galway, Ireland, April 1995.

[46] Lyons, K.W., Duffey, M.R., Anderson, R.C., Product Realization Process Modeling: A Study of Requirements, Methods, and Research Issues, NISTIR 5745, National Institute of Standards and Technology, Gaithersburg, MD, June 1995.

[47] McKay, Alison, Paul Bell, and Bob Young, "Description of Planned Process Data Model," University of Leeds, SERC reference GR/E 04301, 1990.

[48] *Microsoft Project Business Project Planning System*, Version 4.0 User's Guide, Microsoft Corporation, 1994.

[49] *MetCAPP V4 Process Planning Application, Version 4.1 User's Manual*, Institute of Advanced Manufacturing Sciences Inc., Cincinnati, OH 1994.

[50] Molina, Arturo, et al., "Modelling Manufacturing Capability to Support Concurrent Engineering," Concurrent Engineering: Research and Applications **3**(1), 1995.

[51] Nau, Dana S., "Automated Process Planning Using Hierarchal Abstraction," Texas Instruments Technical Journal, Winter 1987.

[52] "NCMS Collaborative Manufacturing Agenda," National Center for Manufacturing Sciences, NCMS Document 0040RE96, May 1996.

[53] National Electronics Manufacturing Framework Committee, "Electronics Manufacturing Technology Roadmaps - and - Options for Government Action", Electronics Industry Association, December 1994.

[54] Ostic, James K., et al, "Integrated Processing and Manufacturing Plant Simulation Using the Process Modeling System," Los Alamos National Laboratory, http://coyote.lanl.gov/PUBLICATIONS/la_ur_94_3574.html, July 5, 1996.

[55] Palmer, Gareth John, "An Integrated Approach to Manufacturing Planning," A thesis submitted to the University of Huddersfield in partial fulfillment of the requirements for the degree of Doctor of Philosophy, England, February 1994.

[56] Ray, Steven R., "A Modular Process Planning System Architecture," Presented at the 1989 IIE Integrated Systems conference, Atlanta, Georgia, November 1989.

[57] Ray, Steven R. (editor), "Proceedings of the 1993 Industrial Process Planning Workshop," NIST Interagency Report 5284, Gaithersburg, MD, October 1993.

[58] Raytheon, "Integrated Process Planning/Production Scheduling," http://agile.cimds.ri.cme.edu/, July 5, 1996.

[59] Raytheon, “Requirements Definition for: MADE - Manufacturing Simulation Driver Draft A,” DARPA MADE program, Administered by Wright Laboratories, Wright Patterson AFB, Contract: F33615-96-C-5609, July 1996.

[60] Rogers, K. J. et al., “The Use of Semantic Networks to Support Concurrent Engineering in Semiconductor Product Development,” University of Texas at Arlington, 1994.

[61] Seesing, P.R., “Ensuring Smooth Sailing: An Overview of Project Planning and Control,” 1995 IEEE International Professional Communication Conference, IPCC 95 Proceedings, 1995.

[62] Semiconductor Industry Association, “The National Technology Roadmap for Semiconductors”, Draft Rev. 2, October, 1994.

[63] Seth, A., “Workflow Automation: Applications, Technology, and Research, Tutorial Notes,” SIGMOD Conference, California, May 1995.

[64] Siegel, Jon, CORBA Fundamentals and Programming, ISBN 0471-12148-7, Wiley, 1996.

[65] Smith, Jeffery S., Joshi, Sanjay B., Message-based Part State Graphs (MPSG): A Formal Model for Shop Floor Control*, 1994.*

[66] Taha, Hamdy A., Simulation Modeling and Simnet, Prentice Hall, 1988.

[67] The Web Site for ProcessModel (ProModel), http://www.processmodel.com/, July 5, 1996.

[68] Uschold, M., King, M., Moralee, S., Zorgios, Y.,” The Enterprise Ontology,” http://www.aiai.ed.ac.uk/~enterprise/enterprise/, July 5, 1996.

[69] Wang, Hsu-Pin, “Process Planning Knowledge Acquisition and Representation,” University of Iowa Project Overview, September 1990.

[70] Watson, G. H., Business Systems Reengineering: Managing Breakthrough Changes for Productivity and Profit, John Wiley and Sons, Inc., 1994.

[71] Workflow Management Coalition Members, Glossary: A Workflow Management Coalition Specification, Belgium, November 1994.

[72] Yu, E.S.K., Mylopoulos, J., “Using Goals, Rules, and Methods to Support Reasoning in Business Process Reengineering,” in Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences. Vol. IV: Information Systems: collaboration Technology Organizational Systems and Technology, 1994.

[73] Zhang, Hong-Chao and Alting, Leo, Computerized Manufacturing Process Planning Systems, Chapman & Hall, 1994.

Glossary

Administrative/Business Extension - a grouping of requirements capable of describing the policies, priorities, and rules pertaining to a company. Requirements such as business practices and manufacturing rules would be included here.

Aggregate Resource/Process Extension - a grouping of requirements used to describe the characteristics of a group of resources.Concepts such as parallelism and implicit resource association are included here.

Business Practices Model (Business Model) - a model describing the rules, strategies, and policies of a business. This may include information such as safety regulations and quality specifications.

Business Process Re-Engineering (BPR) - the fundamental re-thinking and redesign of business processes to achieve dramatic improvements in critical measures of performance such as cost, quality, service, and speed.

Core Requirements - the most basic, fundamental requirements inherent to all processes. To represent process, it is either critical that these requirements be included, or the requirements are so common that every application either explicitly or implicitly acknowledges them. (e.g. time, resource, activity)

Critical Path - the sequence of tasks in which delay in the start or completion of any task in the sequence will cause a delay in completion.

Application Requirements - requirements only relevant within specific applications (e.g., dynamic rescheduling for the production scheduling application).

Enterprise Engineering - that body of knowledge, principles, and disciplines having to do with analysis, design, implementation and operation of an enterprise. Enterprise engineering views the enterprise as a system of cultural, process, and technology components that interact to accomplish strategic objectives and goals. The EE process addresses the critical aspects of how to design and improve all elements associated with the total enterprise through the use of engineering and analysis methods and tools to more effectively achieve its goals and objectives. Enterprise engineering is very similar to business process engineering (defined earlier).

Forecast and Customer Orders Model - a model describing the expected and actual orders for a product.

Functional Requirements - the activities or behaviors relating to a process that the process specification language must be able to support. Some examples are exception handling, dispatching rules, and deadline management.

Inventory Model - a model describing the current or projected amount of a resource available.

Language Grammar - a system of rules for speaking and writing a given language usually dealing with the forms of words and their arrangement in sentences.

Manufacturing Rules Model - a model describing the accepted practices for manufacturing operations. This may include information such as standards data.

Notation - the use of signs or symbols to represent words, quantities, etc. Some example where notations are commonly found are Petri-nets, Pert charts, flow charts, and AND/OR graphs.

Outer Core Requirements - pervasive, but not essential, requirements for describing processes common to most applications.(e.g. temporal constraints, resource grouping, alternative tasks)

Planning/Scheduling/Quality/Analysis Extension - a grouping of requirements required to perform analysis and quality checking. Requirements such as process performance measurements and resource monitoring and feedback are included here.

Extensions - groupings of related requirements, common to some, but not all, applications that together provide an added functionality (e.g., analysis aspects, real-time/dynamic aspects, administrative aspects).

Process - one or more tasks. In general, a process is something that acts over time to change the attributes of objects. A process has three characteristics: a duration, an action, and resources.

Process Characterization Model (PCM) - a model describing the behavior and capabilities of a process independent of any specific application. An example would be a numerical model capturing the dynamic behavior of a process.

Process Intent Extension - a grouping of requirements used to describe the functions and goals of processes. Requirements describing a process' purpose and other decision rationales are included here.

Process Planning - the development of a set of instructions which describe a linear or non-linear sequence of tasks to achieve a specified goal. These instructions can also specify what raw material or components are needed to produce a product, and what tasks are necessary to transform those raw materials into the final product. Process planning is therefore the task of precisely specifying how to manufacture a particular product to its technical data package. As such, process planning forms the link between design and manufacturing.

Process Specification Schema - a schema used to represent the requirements needed to describe a flow of processes. This defines the data structure for the process specification language.

Process Specification Language (PSL) - a language with which to specify a process or a flow of processes, including supporting parameters and settings. This may be done for prescriptive or descriptive purposes and is composed of a schema, a grammar, and one or more notations.

Product Realization Process (PRP) Modeling - computer-interpretable description of the human and machine activities and their interactions required to realize a product. This may include early concept and configuration design activities, detailed design, prototyping, testing, fabrication, assembly and the many other activities within the scope of the realization process.

Product/Resource Characterization Model - a model describing the behavior and capabilities of a product or resource.

Project Management - the application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations from a project. Meeting or exceeding stakeholder needs and expectations invariably involves balancing competing demands among: 1)cope, time, cost, and quality, 2) stakeholders with differing needs and expectations, 3) all identified and unidentified requirements (needs and expectations).

Real-Time/Dynamic Extension - a grouping of requirements required to provide real-time monitoring of a process. Requirements such as process/resource status and states are included here.

Representational Requirements - the characteristics of a process that the process specification language must be able to implicitly or explicitly represent. Some examples are resources, parallel tasks, and non-machining times.

Production Scheduling - the process of assigning activities to resources over time. It is a decision making process where two kinds of decisions may be taken: 1) Time placement decisions: when should each task start? and 2) Resource allocation decisions: Which resource(s), over time, should each task be executed with?

Simulation - the visual or analytic reenactment of a process. Simulation allows the engineering analyst to assess material movement and inventory, waste generation, and personnel and equipment utilization within an integrated plant environment. By building a simulation model to encompass a large system, process chokepoints and complex interdependencies between subsystems can be identified, and proposed alternatives can be assessed.

Stochastic/Statistic Extension - a grouping of requirements used to describe the random or probabilistic aspects of a process. Requirements such as probabilities of down time and performance variability

Systems Integration for Manufacturing Applications (SIMA) - a Congressionally mandated program to support expanded research, development, and standardization of advanced manufacturing systems integration technologies; development and testing of prototype components and interface specifications for manufacturing systems; application of high performance computing and networking technologies to integrate design, planning, and production processes; and testbeds for achieving cost-effective application of advanced manufacturing systems and networks.

Task - a single, identifiable action.

Workflow - a specification of the information processing tasks at a level that describes the process requirements for information system functionality. Workflow essentially captures methods, tasks, and the related software applications in a computer-interpretable form for modeling, simulation, and execution.

Appendix A: Categories and Associated Requirements

1.0 Core Requirements

1.1 Representational Requirements

1. ad hoc notes and annotations optionally associated with any component of a plan
2. cost data
3. level of effort
4. product (work item) characteristics
5. resource
6. resource requirement(s) for a task (with quantity)
7. simple grouping of tasks
8. simple resource capability/characteristics
9. simple sequences
10. simple task representation and characteristics
11. task duration
12. task executor

1.2 Functional Requirements

1. extensibility
2. resource allocation/deallocation for one or many tasks
3. simple precedence

2.0 Outer Core Requirements

2.1 Representational Requirements

1. abstraction
2. alternative task
3. associated illustrations and drawings
4. complex groups of tasks
5. complex resource characteristics
6. complex sequences
7. complex task representation and parameters
8. concurrent tasks
9. conditional tasks
10. confidence levels

11. constraints
12. date(s) and time(s) and/or duration(s)
13. implicit/explicit resource association
14. iterative loops
15. manual vs. automated tasks
16. manufacturing product quantity
17. material constraints
18. parallel tasks
19. parameters and variables
20. pre- and post-conditions
21. queues, stacks, lists
22. resource categorization and grouping
23. resource location
24. resource/task combined characteristics
25. serial tasks
26. state existence constraints
27. state representations
28. temporal constraints
29. uncertainty/variability/tolerance

2.2 Functional Requirements

1. ability to insert or attach a highlight (milestones)
2. complex precedence
3. convey the ancestry or class of a task
4. deadline management
5. dispatching
6. eligible resources
7. exception handling and recovery
8. information exchange between tasks
9. mathematical and logical operations
10. support for task/process templates
11. support for simultaneously maintained associations of multiple levels of abstraction
12. synchronization of multiple, parallel task sequences

3.0 Extension Requirements

3.1 Administrative/Business

3.1.1 Representational Requirements

1. business practices, rules, constraints
2. configuration management information
3. customer-driven process specification and constraints
4. forecast and customer orders
5. priority attributes
6. representation of the origin of task(s)

3.2 Planning/Scheduling/Quality/Analysis

3.2.1 Representational Requirements

1. analysis characteristics
2. critical task
3. predictive and time-dependent resource availability
4. prescriptive task behavior
5. task/process performance measurement

3.2.2 Functional Requirements

1. co-existence of plans and resolution of conflicts
2. dynamic model modification
3. optimization
4. resource/system/process monitoring and feedback
5. support for validation for entire process plan
6. tracking of changes in the system
7. what-if analysis

3.3 Real-Time/Dynamic

3.3.1 Representational Requirements

1. resource amount and availability
2. resource interruptions

3.3.2 Functional Requirements

1. dynamic model modification
2. event signalling and notification
3. resource behavior during processing time
4. resource/system/process monitoring and feedback
5. tracking of changes in the system
6. track in-progress goods

3.4 Process Intent

3.4.1 Representational Requirements

1. decision rationale
2. intentional dimension of processes, or goals
3. relationship between task and goal and resource and goal
4. task/process purpose
5. value-added attributes

3.4.2 Functional Requirements

1. access to past decision rationales

3.5 Aggregate Resources/Processes

3.5.1 Representational Requirements

1. characteristics of groups of resources
2. implicit task association
3. parallelism

3.6 Stochastic/Statistics

3.6.1 Representational Requirements

1. descriptive manufacturing/performance variability
2. probabilities of down times
3. stochastic properties
4. uncertainty of sequences

3.6.2 Functional Requirements

1. account for randomness

2. stochastic functionality

4.0 Application Requirements

4.1 Production Scheduling

4.1.1 Representational Requirements

1. production type data

4.1.2 Functional Requirements

1. dynamic rescheduling

4.2 Process Planning

4.2.1 Representational Requirements

1. clamping surfaces
2. datums and offsets
3. features to be machined
4. production type data

4.3 Simulation

4.3.1 Representational Requirements

1. queue entry and exit rates

4.4 Enterprise Engineering and Business Process Reengineering

4.4.1 Representational Requirements

1. conceptual entities, such as ideas or knowledge

4.5 Workflow

4.5.1 Representational Requirements

1. manual vs. automatable tasks

4.5.2 Functional Requirements

1. invoked tool capability
2. support specifications of task structure (control flow)

4.6 Project Management

4.6.1 Representational Requirements

1. work breakdown ids