Modeling of an Integrated Process Planning System

Qianfu Ni[1], Prasad K. D. V. Yarlagadda[2] and Wen Feng Lu[3]

[1]Queensland University of Technology, q2.ni@qut.edu.au
[2]Queensland University of Technology, y.prasad@qut.edu.au
[3]National University of Singapore, mpelwf@nus.edu.sg

ABSTRACT

The various computer-aided systems in many small and medium sized enterprises (SMEs) work independently at present. This leads to broken information flows and fragmentized business processes, which are the main obstacles to the improvement of business performance. Application integration promotes information sharing, streamlines business processes, and motivates internal collaboration. Moreover, the integration of standalone applications can make use of existing resources and minimize impacts on the current business practices. Therefore, it is an economical approach for SMEs to integrate standalone applications for optimized business performance. This paper presents the modeling of an integrated process planning system, which is intended to assist SMEs to achieve concurrent engineering in design, process planning and manufacturing. Firstly, the integrated system architecture is proposed by addressing the generality and reusability of functional components and collaboration support. Then, a concurrent integration model, which enables different people with different disciplines to work collaboratively, is detailed to tackle the issues of heterogeneous integration and information consistency. Finally, a process planning framework is presented, and manufacturing resource model, part information model and rule-based inference model are elaborated. The development of a prototype system is also briefly presented.

Keywords: Collaborative Process Planning, Integrated Manufacturing, Design for Manufacturing.

1. INTRODUCTION

Application integration is a promising means to assist companies to improve business performance and shorten business cycle [1]. Ultimately, application integration is achieved by building up a fundamental infrastructure to connect various information islands. It provides a mechanism to share information among applications for different activities, departments with different functions and people with different concerns [12]. As a result, information flows are streamlined and consistent so that integral information is made available for better decision-making. The successful cases have shown that business processes can be streamlined by integrating various applications and resource utilization can be optimized to achieve the most optimal performance and the maximum profit. The gains from an integrated system can be significantly improved compared to the situation where applications work independently [5]. The improvements occur in overall business effectiveness by enhancing internal collaboration and coordination, rather than simply in individual department [8].

SMEs usually act as the partners of large companies and operate in a make-to-order manner. As the product market lifecycle becomes shorter, SMEs have to improve their productivity in order to rapidly respond to large companies. At present, many SMEs have adopted some computer-aided systems to assist different business activities. However, these systems work autonomously. Individual systems have their own databases respectively and employ dedicated models to organize and manage data in proprietary manners [7]. This leads to the inconvenient and inefficient interaction between systems and the difficulty to share and exchange data between these systems. Due to the coexistence of multiple copies of the same piece of information at different locations, inconsistency is a potential issue and security management also becomes difficult. Communication between people for different departments is inefficient and error-prone. Therefore, application integration is drawing the attention of SMEs.

However, solutions in the market, such as enterprise resource planning (ERP) systems and product data management (PDM) systems, are usually developed for large companies to achieve advanced planning for batch production, sophisticated design document management, and geographically dispersed collaboration. The license fees and implementation cost are usually the main obstacles for SMEs to adopt these software packages for the integration of various functional departments. After the evaluation of these software packages, in addition to some functional mismatches, one of the common comments from these SMEs is that they can not afford to have professionals to manage and maintain these kinds of software systems. Similar observations were also identified by the requirement analysis of the planning small-medium enterprise network (PLENT) project in the ESPRIT program carried out by the AMICE consortium [11]. This paper presents the modeling of a collaboration-supported process planning system integrated with design and manufacturing to streamline the process from design, process planning to manufacturing.

2. INTEGRATED SYSTEM ARCHITECTURE

System architecture, which conceptually characterizes a software system, is a critical factor to the success of application development and integration. System architecture design is a process to determine the system structure, define core components and identify common functions. From the structural viewpoint, it defines a way for decomposing a system into various interrelated building blocks [2]. From the functional viewpoint, it identifies useful design patterns and supports the patterns by providing abstract layers [4]. Design patterns provide unified approaches to solve similar problems and make application integration and development much more efficient. Therefore, system architecture design is a critical step in application integration. Well-designed system architecture should provide a solid common foundation and enable the focus of system design activities on domain functions with less work on fundamental functions. With these concerns incorporated, the system architecture has been developed as shown in Fig. 1 by adopting the client/server structure.

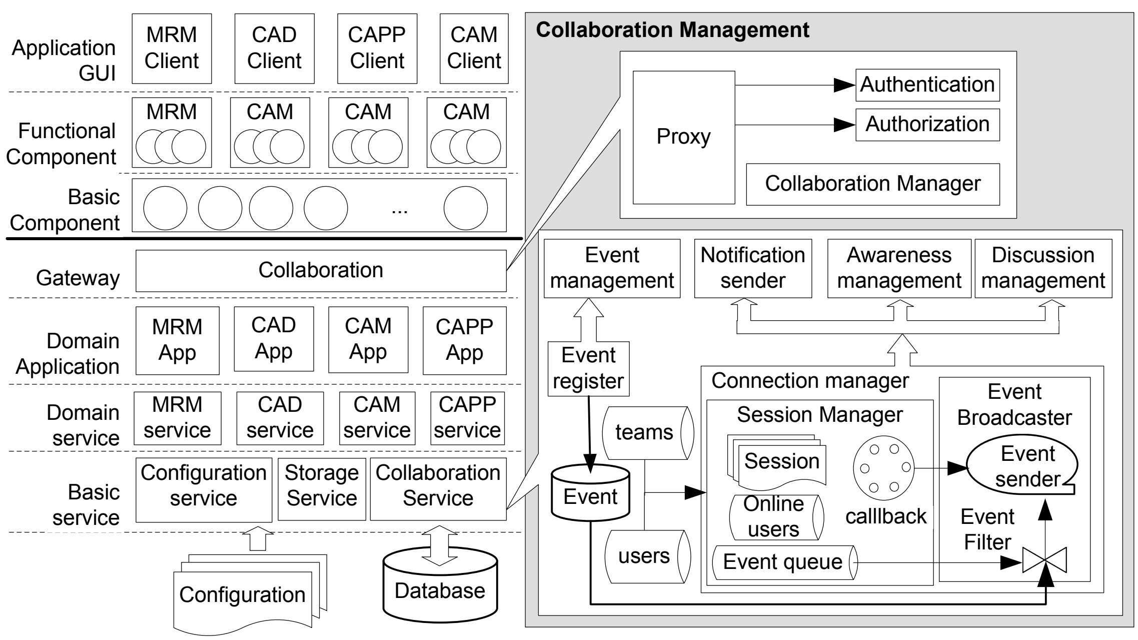

Fig. 1. System architecture.

To achieve better flexibility and reusability, the server is clustered into various layers. At the bottom are basic services, which act as a common foundation for the development of domain-related functions. Domain-related functions are organized into two layers: domain services and domain applications. A domain service provides functions to support individual actions. Here, an action is referred to as an elemental work that can be finished continuously in a short time. An activity is defined as a collection of actions which can be taken at different times according to a business scenario. An activity is performed to fulfill a business objective. A domain application is a component that provides a set of interrelated functions to fully facilitate a business activity. It can be understood as an assembly of functions provided by various domain services based on a business scenario. For example, in process planning, the process planning service provides individual functions for retrieving manufacturing views and process plans, and managing associations between parts and process plans. The process planning application may combine various functions provided by the process planning service and other services to help process planners to do process planning. In this system architecture, information sharing between different applications is achieved at the domain service level. In other word, the system architecture motivates individual applications to acquire shared information managed by other applications via the domain services rather than direct access to the system database. The significance is that the correctness and effectiveness of provided information is ensured as domain services take care of data maturity, integrity and validity. It also assures that information consistency and security can be well managed. To maximize the sharing ability, GUI components of the clients are categorized into three categories: basic components, functional components and application components. Basic components provide fundamental capabilities for presenting data, interacting with users without awareness of domain applications. A functional component is a combination of basic components and other functional components as a big building block with interfaces for communicating with other components. Function components have the ability to communicate with domain applications, domain services and basic services. The proposed system architecture encourages the design of functional components according to the requirements of different applications rather than the needs of individual applications. In this way, functional components can be shared by different applications to reduce development efforts and make the environments of different applications coherent. The architecture is aimed to make basic services, collaboration management, basic components and functional component generic and sharable to different applications.

As shown in the right side of Fig. 1, collaboration support is achieved by two components: collaboration service and collaboration management. The collaboration service provides fundamental functions for managing users, teams, sessions. It also provides basic collaboration functions, such as event management, notification dispatching, awareness management, online discussions and operation synchronization. The event management enables individual users to subscribe the events that he/she is interested in. It aims to enable a collaborative environment to be personalized. The awareness management is a function to support awareness which intends to give collaborators the feeling that network-based collaboration is much similar to the collaboration in physical world by bringing related events to users, such as online users, ongoing tasks, pending tasks and plan release. When a group of users are doing real-time collaboration, unpredictable conflicts could happen if two users are doing operations at the exactly same time. The operation synchronization provides the capability of synchronizing user operations to prevent conflicts. Being built on the top of the collaboration service, the collaboration management has the ability to manage multiple collaboration instances which are involved by different groups of people respectively. The proxy is the entry point of the server. All client accesses are directed to the proxy. The proxy plays three roles: 1) authentication to check whether a user is the promised user; 2) authorization to verify whether a user has a privilege to do certain operation; and 3) forwarder to redirect accesses to collaboration manager or other services. If it comes from the client being involved in online collaboration, the access is redirected to the collaboration manager. The collaboration manager can have the opportunity to coordinate collaboration before invoking domain services or domain applications. The accesses that come from the client which are working standalone are directly forwarded to a corresponding service for processing.

3. CONCURRENT INTEGRATION

In essence, the integration aims to: 1) share and exchange information for better decision-making; 2) enable downstream needs to be considered at an early stage to minimize errors; and 3) prevent unnecessary duplicated work to achieve high working efficiency [13]. To shorten the cycle from design to manufacturing, it is necessary to integrate design, process planning and manufacturing to achieve real-time collaboration and multitasking. Designers, process planners and NC programmers can work collaboratively and simultaneously to evaluate manufacturability, discuss facilities and cutting parameters for machining operations or improve designs for better manufacturability. By addressing these needs, the concurrent integration model has been developed as shown in Fig. 2.

As traditional CAD/CAM systems do not support the client/server structure, most processing functions have to be located at client sides and information generated by these systems can only be stored on local computers. In this situation, it is inevitable to have multiple duplications of the same piece of information. However, in real-time collaboration, consistent information should be maintained for each collaborator. Thus, an effective mechanism needs to be provided to synchronize the information in local databases and the central database. In the proposed integration model shown in Fig. 2, solid model of parts and NC programs to generate parts are stored as unstructured documents in the central database. At the same time, feature-based information of parts is also captured and stored in the central database based on a structural information model. Similarly, process plans and CAM parameters are also stored in the central database. In real-time collaboration, a collaborator, called session controller, initializes a collaboration session and specifies a part to be worked on in the session. Meanwhile, the solid model of the part in the centralized database is downloaded to the client and saved as a temporary file. The temporary part file can be automatically loaded into the CAD/CAM system. Then, other online collaborators with different concerns can be invited to join the session. Similarly, the part model is downloaded to each client and loaded into local CAD/CAM systems. When all collaborators join the collaboration session, they can collaboratively carry out works based on the part from different aspects, such as modeling, process planning, verifying manufacturability, NC programming or evaluating assembly relationships in a real-time manner. During collaboration, the updated part information, process plans, CAM parameters are stored to the corresponding temporary storages on the server side rather than the centralized database. The updated part model and NC program are still store on each local computer. Before closing the collaboration session, part information, process plan and CAM parameters in the temporary storage are updated to the central database. At the same time, the collaborator authorized by the session controller checks in the updated part model and/or NC program to the server. The temporary part model and NC program on other local computers are disposed. In such a way, only information in the centralized database is deemed as legal information for sharing and thus information consistency is achieved.

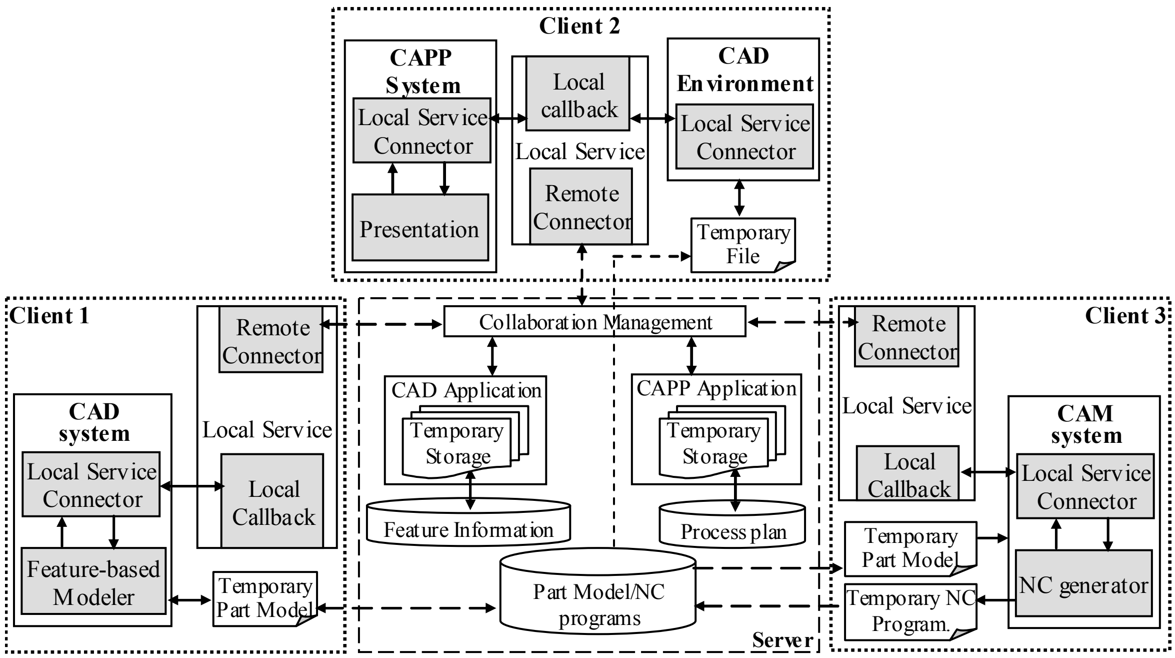

Fig. 2. Integration model.

In real-time collaboration, changes resulted from one client need to be reflected to other clients immediately. To achieve this goal, clients should have the ability to simultaneously communicate with the remote server. Due to the limitation of development toolkits provided by traditional CAD/CAM systems, it is difficult to enhance the ability of traditional CAD/CAM systems to directly communicate with the remote server because of some factors, such as the difference of platforms or different programming languages. However, it is feasible to provide traditional CAD/CAM systems with the ability to communicate with other local applications because they work on the same platform. In the proposed integration model as shown in Fig. 2, the local service connector, which can be developed using the toolkit provided by traditional CAD/CAM systems, is introduced to enable communication between traditional CAD/CAM systems and the local service. The remote connector, which is developed in a language compatible with the server, can create a communication channel between the local service and remote server. In this way, traditional CAD/CAM systems can communicate with the remote server with the help of the local service connector. For example, if a user at client #1 requests to create a new feature, the local service connector will send an event with necessary information to the local service. The local service then forwards the event to the remote server through the remote connector. Next, the remote server broadcasts the event to the feature-based modelers on other clients via callbacks on the clients. The feature-based modelers on each client can replicate the operation and users at each client can see the changes immediately. Therefore, the proposed model is capable of supporting heterogeneous integration. Ultimately, the model provides a means to integrate the process planning system with traditional CAD/CAM systems and support real-time collaboration between process planners, designers and NC programmers. It also provides a mechanism to share information between design, process planning and NC programming. Meanwhile, the consistency of part models, process plans, CAM parameters and NC programs can be well maintained.

4. MANUFACTURING RESOURCE MODELING

In enterprises, manufacturing resource information is widely shared by different business processes for different purposes at different stages of product development [10]. However, resource management capability, if it is really provided by CAD and CAM systems, is very limited. These systems only manage some of manufacturing resources for internal use without the attention paid to the needs of other applications. Moreover, traditional CAD and CAM systems are standalone and must be installed to individual computers. It causes the coexistence of manufacturing resource information on different computers. To achieve the integration of design, process planning, process planning and manufacturing, it is imperative to represent manufacturing resources in a unified way and manage the information in a centralized database. Such a manufacturing resource model should represent different manufacturing resources in a coherent way so that different subsystems for different purpose and different people with different concerns can have consistent understandings to various manufacturing resources. To achieve integrity, a comprehensive data structure needs to be developed to accommodate as much information as possible to characterize every aspect of manufacturing resources and relationships between various resources. From the enterprise viewpoint, a manufacturing resource model should be able to represent the production organization and knowledge about the resource utilization.

By taking these concerns into consideration, the manufacturing resource model shown in Fig. 3 has been developed. It clusters manufacturing resources into three levels according to their roles in decision-making, which are organization level, facility level and knowledge level. The organization of workshop is the projection of the production management practiced by enterprises. It has to be decided based on business strategies and the characteristics of products. Therefore, the organization of workshop can vary from one company to another. Some companies organize workshops according to part types, such as spindle workshop and gear cutting workshop. Others may organize the workshop based on manufacturing processes, such as milling workshops and grinding workshops. Process plans generated should be aligned with the practice of production management characterized by the organization of workshop. Therefore, the model is developed with the capability to represent workshop organization as knowledge for decision-making in process planning. At the facility level, various facilities, including machine tool, cutting tool, fixture and inspection tool, are represented. At this level, facilities are mainly characterized by general information, technical specifications and capabilities, as shown in Tab. 1. Capabilities are basic information in facility selection. Technical specifications are constraints for further refining the selection of facilities. The relationships of facilities also act as the constraints of facility selection. For example, the relationship between machine tools and cutting tools indicates whether a cutting tool can be installed to a machine. While selecting cutting tools, this relationship should be checked to make sure that the cutting tool being considered can be installed to the machine. At the knowledge level, rules are managed for evaluating the suitability of cutting tools with respect to part materials, selecting cutting parameters, determining machining allowance and estimating machining times.

Material information is commonly shared by design, process planning and NC programming. Materials are characterized by physical properties and chemical properties. Designers should consider material physical properties, such as tensile strength, and select proper materials for parts to ensure the performance and life of parts and products. From the concurrent engineering viewpoint, material machining performance should also be considered. In processing planning, material chemical properties must be considered in the selection of cutting tools. In the machining process, high chemical affinity between the materials of a cutting tool and a part can cause sticking, even chemical changes, due to high temperature. This can dramatically decrease the quality of machined surfaces. The cutting tools made of tenacious materials are recommended to cut hard and brittle materials to reduce the chipping possibility of cutting edges. For tenacious part materials, cutting tools made of hard materials should be selected to achieve a high material removal rate by using high cutting speeds. In nature, cutting tool selection is a knowledge-intensive decision-making process. However, no sound theoretic model exists to represent the relationships between cutting tool materials and part materials to guide the decision-making. This paper proposes two relationships between part materials and tool materials to represent this knowledge, which are compatible relationship and repellent relationship. The former is the one which indicates that tool materials are highly recommended for associated part materials. The later implies that tool materials are not proper for associated part materials.

The rules for selecting cutting parameters and determine machining allowances should be semantic and can be easily composed. These rules play a role to support decision-making in process planning. They act as a knowledge base to inherit the experience of senior process planners to assist junior process planners. The rule representation will be further discussed late.

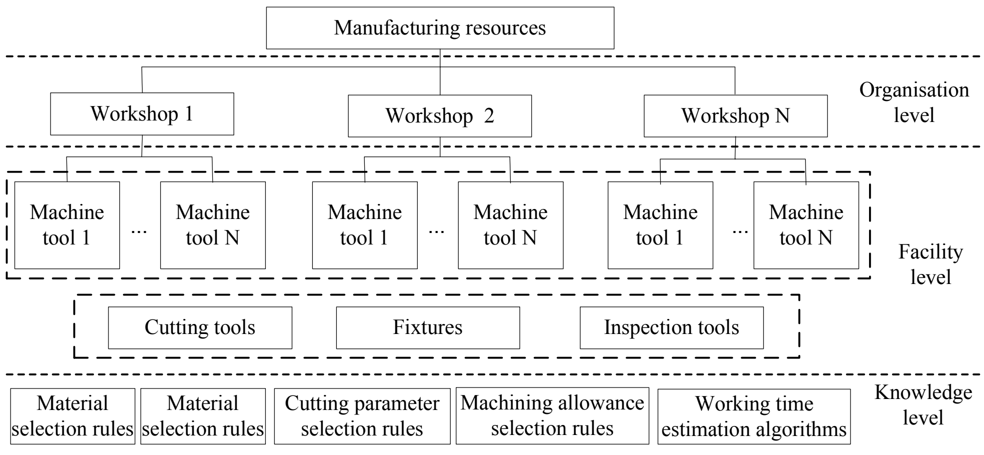

Fig. 3. Integration manufacturing resource model.

Facilities	*General information*	*Technical specifications*	*Capabilities*
Machine tool	id, title, model, vendor	worktable size, economic accuracy	feeding direction, machining operations and machining types (rough, semi-finish, finish and ultra-precision)
Cutting tool	id, title, vendor, chip code	Shank dimension, edge angle and material, etc.	Applicable features, machining types (rough, semi-finish, finish and ultra-precision)
Fixture	id, title	Clamping force, clamping mechanism	Applicable part types (rotational, non-rotational), positioning feature, clamping elements
Inspection tool	id, title	deviation	Measuring range, measuring types (distance, length, diameter, angle, shapes and position, etc)

Tab. 1. Facility specification and capabilities.

5. PART INFORMATION MODELING

It is widely known that the feature concept in the design context is different from that in the manufacturing context. As a result, design features can not be directly shared by manufacturing. In design, features are usually various positive or negative geometric shapes which are organized using a constructive solid geometry (CGS) tree according to the evolving process [9]. Different design systems manage features in different formats. To support process planning, in addition to geometric information, other information, such as tolerance, surface roughness and material are required for decision-making. Some overall information about parts and products, such as part number, part name and GT code, is also necessary for the management of part information and process plans [6]. In this paper, a feature-based part information model has developed to represent part information for sharing. The proposed model classifies features into assistant features, form features and tolerances features. Assistant features include reference direction and reference planes. A reference direction defines a direction along which some machining features exist. Along a direction, multiple planes can exist at different locations, on which machining features seat. These planes are represented as reference planes. Assistant features, which are abstract and virtual, are mainly introduced to facilitate the definition of form features and ease the processing of feature information. Form features are defined based on reference planes and are classified into main features and auxiliary features. Main features are the ones that primarily determine the characteristics of the process plan of a part. Auxiliary features are the ones that are attached to main features, such as chamfer and keyway, etc. An auxiliary feature can not exist without the existence of the corresponding main feature. Auxiliary features are attached to one of the surfaces of main features. In general, auxiliary features are not decisive to the characteristics of the process plan of a part. The model further characterizes form features by using feature surfaces. Basically, there are three types of feature surfaces, which are conical surface, cylindrical surface and flat surfaces. For example, a non-through hole with a flat bottom is shaped by a cylindrical surface and a flat surface while a non-through hole with a conical bottom is shaped by a cylindrical surface and conical surface. In addition, multiple form features can be associated together as a composite form features. In process planning, a composite feature is considered as a whole whenever it is possible. Tolerance features are categorized into shape tolerances, position tolerances and dimensional chains. Shape tolerances are associated with the surfaces of a feature. It implies that different surfaces of a feature can have different shape tolerances at the same time. Position tolerances provide information about the positioning requirement of a feature related to other features. A dimensional chain represents a dimensional relationship between form features.

6. DECISION-MAKING LOGIC MODELING

Process planning is defined as the function within a manufacturing facility that establishes the processes and process parameters (as well as those machines capable of performing these processes) to be used in order to convert a part from its initial form to a final form predetermined (usually by a design engineer) on a detailed engineering drawing [3]. The deliverables of process planning are process plans, which are usually represented as process sheets and can be further classified into routing sheets and operation sheets. Essential information in process plans includes processing methods, features to be machined, cutting parameters and dimensions to be achieved as well as machines, cutting tools, fixtures that can be used for each operation. A framework of process planning has been developed, as shown in Fig. 4. It organizes process planning into different interrelated steps, which are selection of machining operations, generation of routings, selection of machines, sequencing to operations, selection of fixtures, selection of inspection tools, selection of cutting tools, determining machining allowance, selection of cutting parameters and estimating machining time. In this way, individual steps can be collaboratively carried out respectively.

In the framework, capability is an important concept. For the system to work efficiently and share information effectively between different subsystems, a capability coding schema has been developed to represent capabilities of various facilities. As an example, the machine capability coding schema is illustrated in Fig. 5. The machining capability is defined based on feeding directions. It implies that a machine can have different capabilities in different direction. In the schema, the first character indicates main manufacturing process, such as machining or heat treatment. The second and third characters indicate feeding directions. The characters from 4 to 7 provide information about operations types and subtypes. The characters from 8 to 11 define the targeted feature of the operation. The last character represents the operations precision. Based on this capability representation, various feeding directions can be defined for a machine and a set of capabilities codes can be associated with each direction to define its capabilities. The proposed code schema provides the three major benefits: 1) providing a means to structurally and semantically represent facility capabilities; 2) providing effective linkages between facilities, operations and features to support the integration of design, process planning and manufacturing; and 3) simplifying decision-making mechanism by converting the processes of operation selection and facility selection to the process of semantic code matching and analyzing. It also improves the system performance, especially in the real-time collaboration situation, by reducing the bulk of information to be transferred in an integrated environment.

Rule is another important concept in the framework. The rule-based inference model has been developed by taking the advantage of XML technology. As shown in Fig. 6, rules are configured in XML, which is linguistic and legible. A rule consists of condition and conclusion. A condition is a logical expression which comprises a set of operands linked by mathematical operators and logical operators. Operands in the condition expression can be constants, entity attributes, functions and other expressions. The entities, such as parts, features or resources, are referenced using unique keywords which are predefined. The functions represent common information processing logics and are identified by reserved keywords. They improve the efficiency of rule construction and rule evaluation. In the rule evaluation, the configured rules are converted to hierarchical trees, each node of which represents an operator, a constant, a function call or an expression call. Part information, partially generated process plan and resource information are inputs for rule evaluation. They are mainly used to initialize rule evaluation contexts. Units are important to engineering data. The unit analyzer plays a role to unify data units before processing. The XML-based representation makes rules be easily composed. More important, decision-making logics are decoupled from computer programs. The programs work by interpreting the logics defined in separate configuration files. It offers the flexibility to tune rules or change rules for new practices. The system can also be generic and work for different companies by providing different sets of rules.

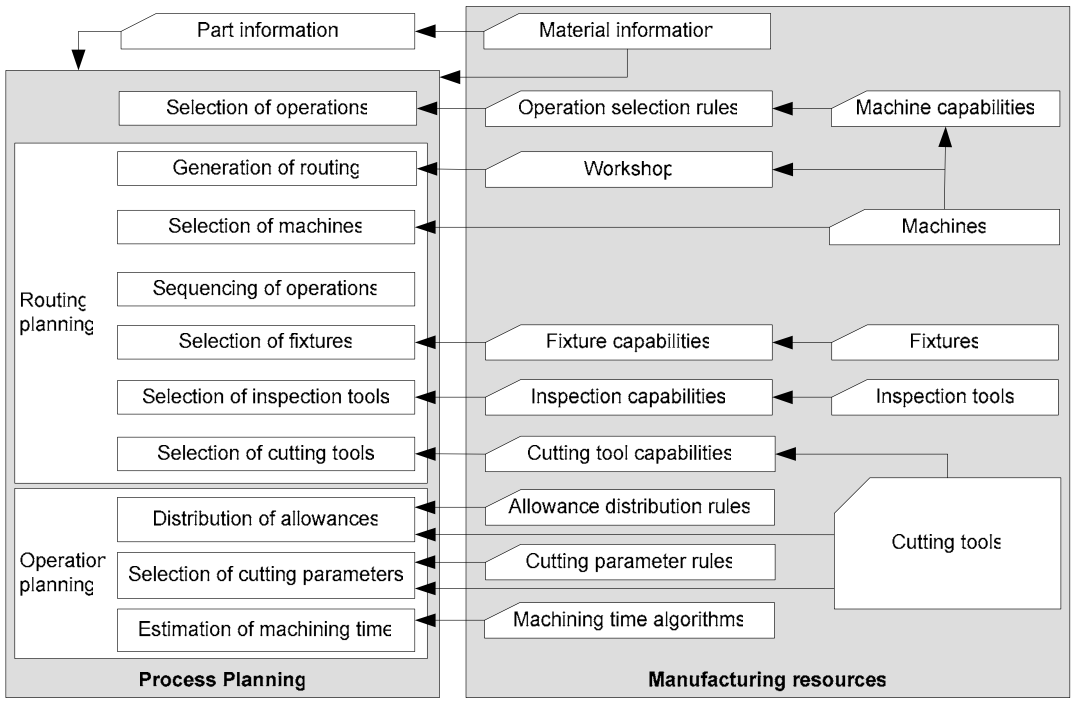

Fig. 4. Process planning framework.

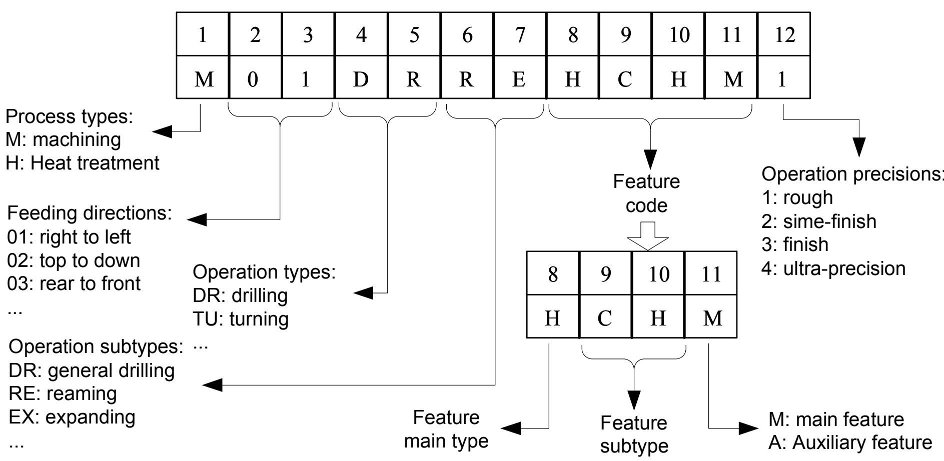

Fig. 5. An example of machine capability coding.

7. A PROTOTYPE SYSTEM

A prototype has been developed as a proof-of-concept system to verify and demonstrate the developed architecture and models. The prototype system is developed based on the Windows platform using the distributed computing technique of component object model (COM)/distributed-COM (DCOM) by Microsoft. COM/DCOM is an object-oriented and component-based technique for developing client/server-based applications. As COM/DCOM is language-independent and provides components with high interoperability, it is suitable for the integration of heterogeneous applications. In addition, COM/DCOM is a native function of Windows platform, which is the dominant platform in SMEs. Microsoft SQL server is adopted as the database server.

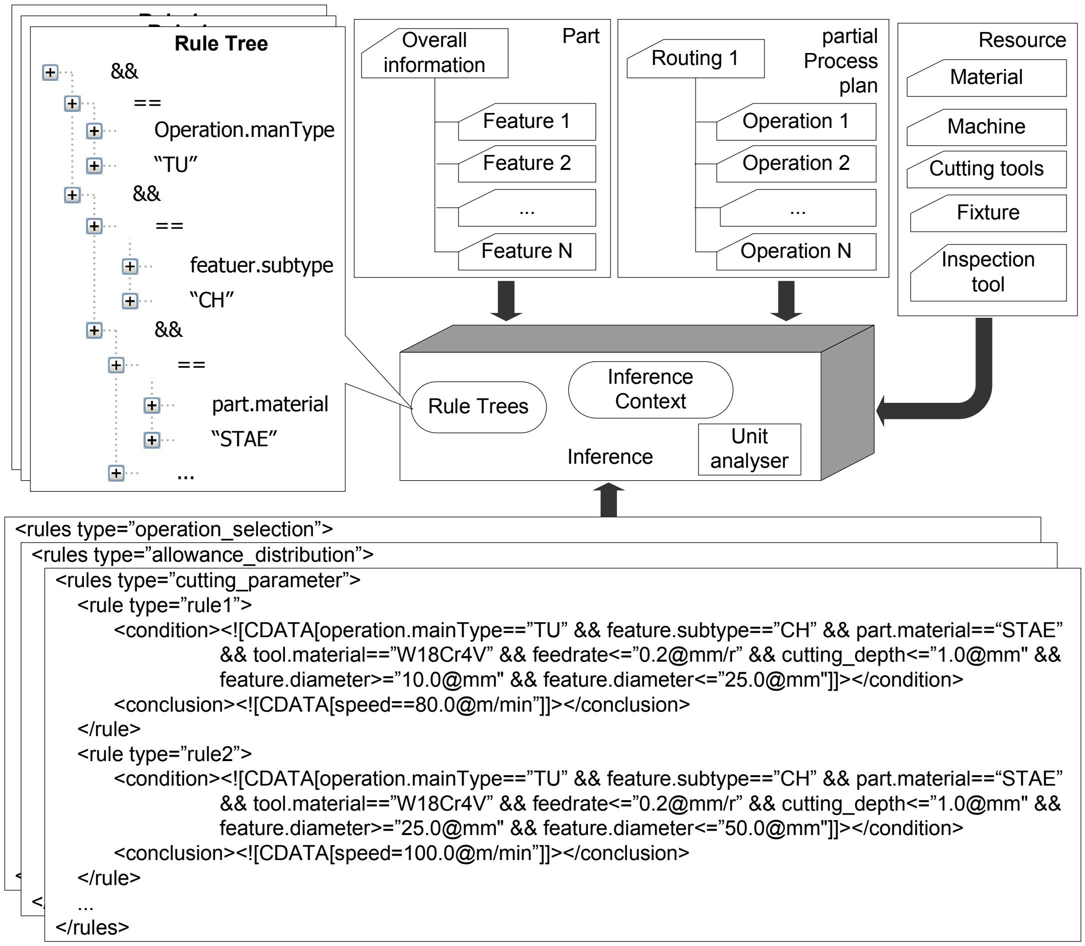

Fig. 6. Rule-based inference model.

In the prototype, the process planning and manufacturing resource management subsystems are developed from scratch. Unigraphics, a CAD/CAM system by Unigraphics Solution, is integrated to support design and manufacturing. To build up a heterogeneous integration environment, different components and subsystems are developed in different languages. All server components are developed as DCOM components in C++. In all the subsystems, the local service and remote connector, as shown in Fig. 2, are developed in Visual J++ and C++ respectively. The local service is developed as a Windows service which is a special type of Windows applications. As Windows services can be automatically started and managed by Windows, they are suitable for connecting other applications. The client applications of process planning and manufacturing resource management and their local service connectors are developed in Visual J++. The clients of process planning and manufacturing resource management are pure graphical user interfaces (GUIs) for information presentation as the information processing functions are developed into domain applications and domain services and deployed on the server side. The feature-based modeler and NC generator are developed based on UFUN, which is a C/C++-based development toolkit provided by Unigrpahics. The service connectors of CAD and CAM subsystem are also developed based on UFUN. As Unigraphics does not support the

client/server structure, the feature-based modeler and NC generator are deployed on client sides. The feature-based modeler can capture and organize design information based on the part information model discussed above.
The integrated system provides the following advantages: 1) designers can simultaneously work with process planners and NC programmers to evaluate manufacturability; 2) process planners can collaboratively create the process plans of parts; 3) NC programmers can discuss the utilization of manufacturing resources with process planners in a real-time manner; and 4) part information and manufacturing resource information can be commonly shared by designers, process planners and NC programmers.

8. CONCLUSION

To address the needs of integrated manufacturing in SMEs, an integrated process planning system has been modeled. The collaboration-enabled and multi-tier system architecture has been proposed for integrating heterogeneous applications and sharing information among design, process planning and manufacturing. By clustering various functional components into different layers, the system architecture provides a means to promote the reusability of functional components, maximize information sharing and, at the same time, enable the consistency and integrity of information to be well managed. Based on the system architecture, a concurrent integration model, which is capable of supporting concurrent collaboration among design, process planning and manufacturing, have been developed with the concerns of information synchronization in the integration of traditional CAD/CAM systems. A level-based manufacturing resource model and a feature-based part information model have been developed to facilitate integration and decision-making. A capability coding schema has been introduced to effectively convert the decision-making in process planning to a process of capability code matching and analyzing. A semantic rule configuration technique has also been designed to provide a flexible way to define rules for decision-making in process planning. A prototype has been developed as a proof-of-concept system to demonstrate the developed system architecture and models.

9. REFERENCES

[1] Chalmeta, R., Campos, C., and Grangel, R., References architectures for enterprise integration, *Journal of Systems and Software*, Vol. 57, No. 3, 2001, pp 175-191.

[2] Doumeingts, G., Ducq, Y., Vallespir, B. and Kleinhans, S., Production management and enterprise modelling, *Computers in Industry*, Vol. 42, No. 2-3, 2000, pp 245-263.

[3] Groover, M.-P., *Automation, Production Systems and Computer Integrated Manufacturing*, Prentice Hall, London, 2000.

[4] Gamma, E., Helm, R., Johnson, R., and Vlissides, J., Design Patterns, Addison Wesley, Holland, 1998

[5] Gunasekaran, A. and Thevarajah, K., Implications of computer-integrated manufacturing in small and medium enterprises: an empirical investigation, *International Journal of Advanced Manufacturing Technology*, Vol. 15, No.4, 1999, pp 251-260.

[6] Huifen, W., Youliang, Z., Jian, C., Lee, S.-F. and Kwong, W.-C., Feature-based collaborative design, *Journal of Materials Processing Technology*, Vol. 139, No. 1-3, 2003, pp 613-618.

[7] Kim, C., An object-oriented information modeling methodology for manufacturing information systems, *Computer & Industrial Engineering*, Vol. 24, No. 3, 1993, pp 337-353.

[8] Levy, P., Bessant, J., Levy, C., Smith, S. and Tranfield, D., Organizational strategy for CIM, *Computer-Integrated Manufacturing Systems*, Vol. 4, No. 2, 1991, pp 80-89.

[9] Li, W.-D., Ong, S.-K. and Nee, A. Y. C., Recognizing manufacturing features from a design-by-feature model, *Computer-Aided Design*, Vol. 34, No. 11, 2002, pp 849-868.

[10] Liu, C., Wang, X., and He, Y., Research on manufacturing resource modeling based on the O–O method, *Journal of Materials Processing Technology*, Vol. 139, No. 40-43, 2003, pp 40-43.

[11] Mezgar, I., Kovacs, G.-L., and Paganelli, P., Co-operative production planning for small- and medium-sized enterprises, *International Journal of Production Economics*, Vol. 64, No., 2000, pp 37-48.

[12] Ortiz, A., Lario, F. and Ros, L., Enterprise Integration--Business Processes Integrated Management: a proposal for a methodology to develop Enterprise Integration Programs, *Computers in Industry*, Vol. 40, No. 2-3, 1999, pp 155-171.

[13] Vernadat, F.-B., Enterprise modeling and integration (EMI): current status and research perspectives, *Annual Reviews in Control*, Vol. 26, 2002, pp 15-25.