

VIRTUAL MANUFACTURING - ONLINE CONTROL OF SHOP FLOOR OPERATIONS

Jayendran Venkateswaran, Maheer Manmohan and Young-Jun Son

Systems and Industrial Engineering Department

Engineering Building #20, Room 111

The University of Arizona

Tucson, AZ 85721-0020, USA

Tel: 1-520-626-9530, Fax: 1-520-621-6555

E-mail: son@sie.arizona.edu, Web site: <http://www.sie.arizona.edu/faculty/son>

ABSTRACT

The primary focus of past research in the field of Virtual Manufacturing (VM) has been confined to the use of Virtual Reality in manufacturing. This paper tackles the subject from the perspective of using the Internet to directly control the machines in real time. The capabilities of VM are extended through this approach, and are found to have applications in such diverse areas such as distance learning and remote shop floor control. A prototype model has been developed in the Computer Integrated Manufacturing Lab at The University of Arizona. The model has six distinct modules: 1) Design Input module, 2) Process Planning module, 3) Code Generation module, 4) Simulation module, 5) Machine Control module, and 6) Integrator module. Functions of the modules in the model are detailed using IDEFØ modeling method. This model has been implemented using Java™, Java™ Servlets, C and Microsoft Access™. Future challenges and applications are also discussed.

Keywords: *Virtual Manufacturing, Shop floor operations, Internet based machining*

1. INTRODUCTION

The Internet is currently used for purposes of information interchange and e-commerce. Textual materials can be presented in the Internet environment using the various authoring tools. Graphics and animation are also used to improve the user's readability of the contents. Currently, online transactions have also become universal, giving rise to a new commerce, e-Business. However, to make this technology more useful for the manufacturing environment, it must be able to remotely control physical entities. This leads to the area of Virtual Manufacturing (VM). The past research in the area of VM (Shukla *et al* 1996) has been from a perspective of using virtual reality, i.e., modeling tools and three-dimensional techniques, to dynamically simulate the various operations. The capabilities of the virtual reality interface make it an ideal tool during a concurrent engineering product-development project. In this paper, we present an approach to extend the capabilities of the Internet to shop floor control. This paper broadens the scope of VM to include remote control of machines in real time (Di Stefano *et al* 1997). The term *Virtual* in our paper implies the manner in which certain manufacturing objectives are met and how the equipment are controlled, and not virtual reality. Specifically, the Internet is used to remotely control the physical operations of any machine in real time (Küchlin *et al* 1997). The increased focus on mass customization of products, distance education programs, and safer work place

practices provides ample possibilities for the practical applications of the technology discussed in this paper (Layne *et al* 1998).

In the next section we give a brief description of the related work in various fields associated with VM. Then we present an overview of our system in Section 3. In Section 4 we discuss the issues involved in the development of the system. A detailed architecture and functional models using IDEFØ method are presented in Section 5. Then, the actual implementation of the technology has been covered in Section 6 in a modular way. A demonstration of the system is presented in Section 7. Section 8 discusses the applications of this system and the directions of future research. Finally, we conclude the paper with our findings.

2. RELATED WORK

Past and current work related with our system is described below. These works have helped steer our pedagogical viewpoints and our development decisions.

2.1. *Computer Integrated Manufacturing*

Before elaborating on the importance and relevance of Virtual Manufacturing, it is imperative to discuss how computers have been used thus far in a manufacturing environment. It is common in the present time to have designers and engineers use software not only to visualize the product that is being manufactured but also to implement a solution. The use of Computer Aided Design (CAD), Computer Aided Manufacturing (CAM), and Computer Aided Process Control, as tools to aid in manufacturing is an accepted fact. Although the automotive and aerospace industries are at the forefront of the development of CAD technology, the evolution of the same can be traced back to the development and use of interactive computer graphics and powerful computers with high processing speeds. A CAD system basically consists of three major components - the operating system, the software specific to the application needs, and the hardware that is constituted by the PC and associated devices. The CAD systems are typically integrated with CAM systems to achieve overall manufacturing goals. The CAD/CAM systems are generally classified into four types: Mainframe-based, Minicomputer-based, Workstation-based and Microcomputer-based systems. The selection of CAD/CAM systems is difficult given the number of vendors in the market who are eager to incorporate features that are both necessary and un-necessary. To overcome this hurdle, a weighting scheme could be used in which the goals and the needs of the company are evaluated, and priority is assigned on that basis. The issues that are discussed include system-related issues, geometric and modeling issues, and the applications supported by the system. The translation of design requirements into process measures is a vital step in manufacturing. The Computer-aided Process Planning (CAPP) approach has been developed with a view to overcome the inherent problems associated with the manual process planning approach. One of the main problems associated with manual process planning is the time and effort that is consumed in the development of plans, and the inconsistency that may arise over a period of time. The cost associated with such a method is also considerable, and may involve lots of skill and experience. Besides the factors such as lesser time and effort, the advantage of the CAPP is its ability to generate alternatives in case of any problem with the original process plan, such as the breakdown and repair of any machine.

The CAPP also helps to produce plans at a high level of accuracy, and also helps to involve other application programs such as cost estimates integration, manufacturing lead times estimation, and work standards. The two main methods used in the CAPP are the variant and generative methods. The variant method essentially draws upon other existing process plans for a

similar part to produce a modified version, which is the process plan for the new part. The strategy here is to produce the process plans for a whole part family, and then to use the fact that parts have similar design characteristics and manufacturing properties to derive the process plan for the appropriate part. The generative approach uses decision logic as the basis for the development of the process plan, and may take data from the part geometry, formulas, and technology-based algorithms as its input. The two major components for these technology-based algorithms include a geometry-based coding scheme, and process logic. The former involves the proper definition of geometric features, tolerances etc., while the latter is more involved with the use of decision tables, and knowledge-based systems (Singh 1996).

2.2. *Virtual Manufacturing (VM)*

New advances in computerized modeling, visualization, simulation, and product data management are making virtual manufacturing a viable alternative to traditional product realization/manufacturing. VM can be defined as the use of computer models and simulations of manufacturing processes to aid in the design and production of manufactured products. VM has been created to provide a modeling and simulation environment so powerful that the fabrication/assembly of any product, including the associated manufacturing processes, can be simulated using the computer. In other words, VM will accommodate the visualization of interacting production processes, process planning, scheduling, assembly planning, logistics from the line to the enterprise, and related impacting processes such as accounting, purchasing and management (Lawrence Associate Inc 1994). The major benefit of a VM environment is that the physical system components (such as equipment and materials) as well as conceptual system components (e.g., process plan and schedules) can be easily represented through software entities that emulate their structure and function (Dessouky *et al* 1995). The close correspondence between the virtual entities and the actual system lends validity to the idea of using simulations carried out in the virtual system to aid decision-making for the real system (Dessouky *et al* 2001).

The vision of Virtual Manufacturing is to provide a capability to "Manufacture in the Computer". In essence, VM will ultimately provide a modeling and simulation environment so powerful that the fabrication/assembly of any product, including the associated manufacturing processes, can be simulated in the computer (Lawrence Associate Inc 1994). This powerful capability would take into account all of the variables in the production environment from the shop floor process all the way up to enterprise transactions. VM will accommodate the visualization of interacting production processes, process planning, scheduling, assembly planning, logistics from the line to the enterprise, and related processes such as accounting, purchasing and management that also impact the production environment.

2.3. *Use of Internet and JavaTM in Manufacturing*

Internet technology has developed rapidly and spread its tentacles across the world in the past decade. The Internet has been a boon because of its ability to connect people across the globe, not only to communicate with one another but also to conduct business. Thus, it is not uncommon to find radios, televisions, cars, and even houses changing hands over the Internet. Industry has also exploited the potential of this technology has also been exploited in a manufacturing environment. Manufacturing companies in the United States have adopted the Internet and B2B initiatives and are making new and better business models. In an enterprise, the various divisions including sales, manufacturing, finance, engineering, etc, are being networked

to achieve better co-ordination thereby increasing productivity and profit. But currently the focus is shifting towards a much higher level of interaction -- the interaction between two factories of the same organization, and even the interaction between the factory, its supplier and its retailer (supply chain integration). These call for the use of the Internet.

On the manufacturing side, the internet has allowed us to create a new environment for supporting implementation of a new concept called distributed manufacturing, which makes distributed remote machining possible (Adamczyk *et al* 2001). This methodology will support remote users and facilities to ensure the performance of manufacturing equipment, quality of operations, and productivity of plants. For example, the performance of a machine or piece of automated office equipment could be monitored and assessed from anywhere in the world. In a manufacturing setup, the use of the Internet enables the networking of machines with one another and with the operators who handle them (Lee 1998).

The programming language that is popularly used for these purposes is Java TM, from Sun Microsystems. It helps to transform the Internet into a medium of secure online and real-time transactions that are independent of the operating system. It provides features for easier implementation of communication across different networks. Java TM can also be combined with a variety of other applications or languages to fulfill the design of the architecture.

Java TM also provides features to interact with the manufacturing databases. Often there is different database for each department within the organization catering to various needs like design, finance, marketing etc. The problem of varying data structure and even database systems is easily solved by using Java TM.

3. OVERVIEW OF OUR MODEL

This paper extends the capability of Virtual manufacturing to the shop floor. This system enables real time control and monitoring of manufacturing equipment by providing instant feedback on the current status of the system to the remote user. Thus it lends itself to extending the capabilities of the Internet by directly controlling the equipment. Further, it opens up opportunities to manage equipment in unmanned and hostile environments.

The software components are integrated in a layered and modular architecture so that the modification and upgradation of individual modules can be performed without affecting the other layers. Figure 1 shows the different software components and sub-components that together define the system. The system consists of three different layers: (i) real shop floor components (ii) the VM Server components, and (iii) the client components. The client side components consist of the web browser, Java TM scripts and Java TM applets. Refer to Section 4.3, Technical Issues, for details regarding all the software used. The client components, using the user interface provided, send in their requests to the web server. The VM Server layer acts as the interface between the client and the actual shop floor components. It interacts with the equipment and/or the equipment controllers in the real shop floor to machine the parts accordingly.

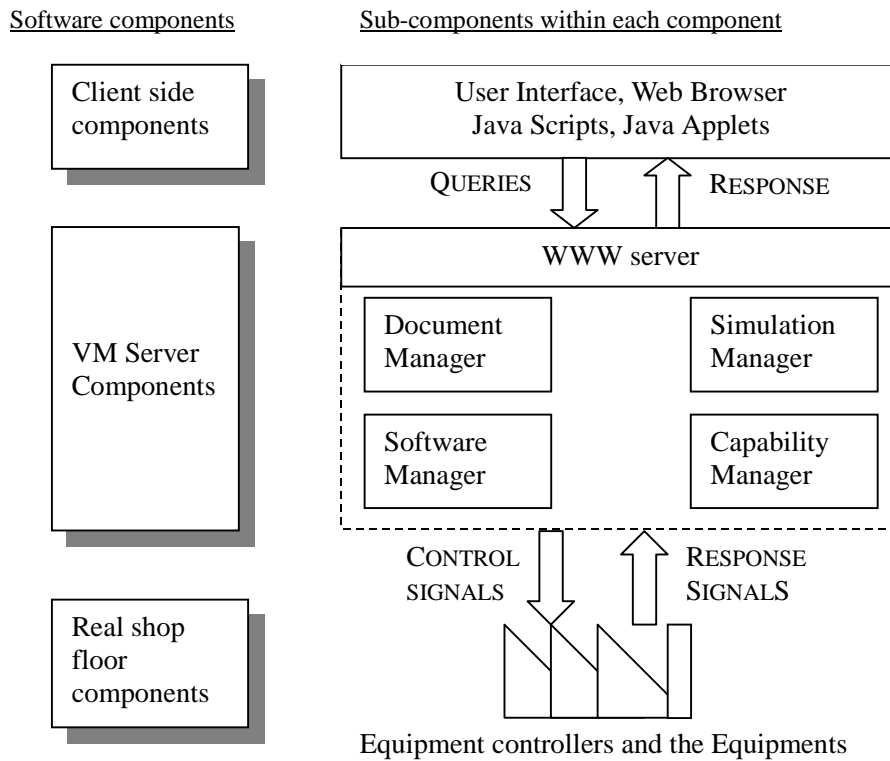


Figure 1. Software components and sub-components of the Virtual Manufacturing System

The various components of the VM Server layer are described in Table 1. More diverse equipment and equipment controllers can be added to the real shop floor layer without modifying the entire system.

Table 1. Software components of VM layer

<i>WWW Server</i>	Enables communication to and from the remote user through the Internet.
<i>Software manager</i>	Manages all the software behind the system. It is essentially a collection of the Java TM servlets, applets, html pages, and other software programs.
<i>Document manager</i>	Manages all the user specific documents involved like the user settings file, design files, NC code files, video clips etc. as required by the system. Also consists of the Database required for storing process plan and simulation results.
<i>Capability manager</i>	Makes available the capability of the system and the shop floor to the remote user.
<i>Simulation manager</i>	Simulates and authenticates the remote user's plans and designs before actually executing in the shop floor. Also contains the simulation configuration files.

Functionally, the system is classified into six independent modules that are residing in the WWW Server: (i) Design Input module, (ii) Process Planning module, (iii) Code Generation module, (iv) Simulation module, (v) Machine Control module, and (vi) Integrator module. The Integrator module is a 'virtual module'. It is not an explicit module; the web server itself implements it.

The User, with the help of the World Wide Web browser, interacts with the Server. The Server helps the user to specify the design of part that needs to be manufactured, generates the process plan for the design, generates the specific control codes for the machines, tests the code through a dynamic simulation, and finally executes the code on the machines. Depending on the type of the input, the Server, through the Integrator module synchronizes the activities of all the other modules to achieve the desired output. The feedback is then provided to the user. All the requests, the intermediate feedback, and the final responses of the modules are stored in a database for the purpose of information interchange between the different modules.

4. ISSUES CONCERNED WITH THE DEVELOPMENT OF THE MODEL

When developing a system for manipulating physical entities in shop floor operations via the Internet, the issues to be considered are copious. They have been narrowed down to three topics and are discussed in this section.

4.1. *Design Issues*

The proposed system has the following functionalities:

- It has a Computer Aided Design module.
- It has a Computer Aided Process Planning module.
- It generates manufacturing instructions (NC code generation)
- It dynamically simulates the machining operations before actual manufacturing.

These modules have been identified based on the basic steps involved in any manufacturing system (Chang *et al* 1998). Each of these functions or modules by itself demands a great deal of research work. The thrust of the paper is not a specific problem; instead it is in the development of a web-enabled integrated system of the various modules. As a result, the bulk of the work was involved in the creation of the system and within the modules themselves, only enough work was done to get the system working as a whole.

For any part to be properly manufactured, a detailed part representation with information pertinent to manufacturing is required. Computer Aided Design (CAD) is the use of computers to design a part and to analyze the design. In the current system, the CAD module is used as a medium for the users to specify their own product designs. Reusable product templates are used to generate customized products. Based on the availability of resources, several basic product designs are established. These are provided to the user as templates to fashion customized products. Boundaries are set to limit the extent of modifications to the base templates.

A detailed process plan contains the route, processes, process parameters, machines, tools, and other instructions required for producing a part. The process plan is of utmost importance as it dictates the cost, quality and rate of production. Computer aided process plans (CAPP) are traditionally based on variant or generative approaches. The variant approach uses standard plans to derive a similar new component. The generative approach creates the process plan automatically for new components. For the current system a semi-generative approach, combining the advantages of both the approaches, is followed.

Using CAD-to-CAM conversion, the corresponding NC codes, cutting parameters etc are determined. With the availability of product templates, corresponding NC code templates are maintained in the database. Customization of the product results in equivalent changes in the machine instructions.

Verification is an important step in any system before actual implementation. Simulation is used as the verification tool. Simulation can be categorized into different levels such as, tool path simulation, cell level simulation, shop floor level simulation, factory level simulation, supply chain simulation, etc based on the level of abstraction. In the current system the tool path alone is simulated dynamically based on the customized machine instructions. The current simulation can be expanded to further incorporate cell level, shop level and factory level simulations.

4.2. Technical Issues

The technical issues revolve around the choice of the software and hardware required for the implementation of the model. This being an Internet based model, the use of Java™ and Java™ Servlets (Hunter 1998) from Sun Microsystems provides a medium for secure online and real-time transactions. Using the Java™ language it is possible to create programs that can be called by any user using a browser and executed locally in the user machine. These Java™ applications are platform independent, which mean they can be run on computers with Win32, OS/2, Solaris or MacOS operating systems. Java's built-in support for networking helps in easier development.

The client-server communications can be handled in a variety of ways. The Hyper Text Transfer Protocol (HTTP) connection or raw socket connection can be used to achieve the same thing. The HTTP connection works even behind firewalls, allows secure communication, and is easy to develop. Sockets offer a bi-directional, sustained connection but often do not run behind firewalls, and require a much more complicated server and also a development of a custom protocol. These factors and the requirements of modularity in the system favored the use of HTTP protocol. The use of Java Web Server™ further eases the development process as it comes with lots of features in implementing the client-server communication.

The Server side programming has been implemented using Java™ Servlets. Earlier, only the Common Gateway Interface (CGI) was used for interfacing external applications with information servers, such as HTTP or Web servers. Servlets are generic server extensions – a java class that can be loaded dynamically to expand the functionalities of the server. The servlets can be used with web servers, where they can take the place of CGI scripts. The use of Servlets over CGI is easily justified (Sun Microsystems *website*). To begin with, they are written in Java, and thus extend all features of the Java™ language. They are portable across operating systems; do not drain system resources; and come with a certain amount of basic initial infrastructure. Servlets are also efficient and scalable. Because servlets run within the web server, they can interact very closely with the server, enabling us to do things that are not possible with CGI scripts (Sunworld *website*)

The Client side programming invariably calls for a user-friendly graphical interface, called the Graphical user interface (GUI). The GUI is developed for the system for easy access for the user to the various modules. The Hyper Text Markup Language (HTML) was used extensively to develop a simple yet powerful GUI for the user to interact with.

The actual low-level machine control logic can be realized more efficiently using C language. The C language provides more features to interact with hardware than Java. The control logic involves serial or parallel port communication or both, depending on the configuration of the equipment available.

All the required information and data regarding designs, NC codes, machine capabilities, process plans etc can be stored in any of the following ways: as files in the server, in a database, or as arrays in the program itself. The last option is very inefficient as it results in a lot of overhead in the program, which would increase exponentially with the number of users. Storing in files eliminates the overhead in the program but does not offer a structured way to store information. The use of a database overcomes both of these obstacles. Any database system can be used for this. The java program requires the Java Database Connectivity (JDBC) TM API to communicate with the database. JDBC is the standard for communication between a Java TM application and a relational database. With the development of the JDBC, it is possible to create secure, three-tier database systems. All of the pertinent information for manufacturing can be simply retrieved from the existing database without much additional overhead.

4.3. *Security Issues*

Security is one of the paramount issues when developing any Internet-based application. The natural vulnerability of web-based applications or websites has encouraged many hackers to carry out attacks on some of the most sensitive and protected installations around the world. Preventing such security breaches and preserving integrity is a pivotal issue for all applications that make use of the Internet. The standard measures to prevent such complications include the use of firewalls and authentication procedures to allow only genuine and authorized users. In this paper, the security issue has been dealt with from a high level standpoint. While the pitfalls associated with security problems are acknowledged, this does not constitute the primary focus of the paper. Rather, it is a small subset of the overall case that we are discussing, and we would like to wrap up further analysis of the same by pointing out some more measures that can be adopted. For instance, a cryptographic security (Molva 1999) measure can be added on to the system without any modifications in the model. Other examples include security mechanisms like authentication, encryption and key exchange techniques.

A necessary-security issue concerning the access of the machines and accessories in the shop floor is considered and dealt with accordingly. Each machine in the shop floor is a shared resource, that is, they are accessed and used by more than a single client. These are the client level shared resources. Within the shop floor the tools, work pieces, buffers etc are shared, that is, they are used by more than one machine. These are the shop level shared resources.

The system must ensure that two activities that use the same-shared resources (client level or shop level) may not occur at the same time. To achieve this, the different activities are separated or mutually excluded. The mutual exclusion can be seen as a condition that specifies that only one activity may use a shared resource at a time to ensure correct operation and results. Small software programs within the system, called critical regions, guarantee that the current client or activity completes its usage of the resource before any other client or activity can have access to the resource being used. The concept of mutual exclusion being the same, the level of implementation to handle the different types of shared resources differs. The client level mutual exclusion is taken care by using Java TM while the shop level is taken care of by C program. These ensure a smooth and safe access to the shop floor for the remote client.

5. FUNCTIONAL MODELING OF THE SYSTEM

A detailed modeling of the activities of the VM system from its operation viewpoint is performed to describe the different occurrences of the activities, by system and subsystem, according to the information or objects that serve to distinguish these occurrences. An activity

model is a structured representation of the information and objects that define a correspondence between the activities occurring in a manufacturing system or environment. The primary focus of activity modeling is on the information and objects that create a unique occurrence of a given activity. A functional modeling tool, Integrated DEFinition Ø (IDEFØ) is used to model the decisions, actions and activities of the system (Mayer 1992, Shin *et al* 2000). The system model is decomposed up to three levels, as shown in Figures 2a, 2b and 3, to get a more detailed look. The Activities are represented by the rectangular boxes. The Inputs to the activities are the arrows connecting to the activity boxes from the left side. Arrows leaving the activity boxes from the right side represent the Outputs of the activities. The Control, a condition that governs the activity performance, is shown feeding in to the activity box from the top. The Mechanism by which the activity is carried out is shown by the arrow to the activity box from the bottom.

Figure 2a shows the Top-level or A0 diagram or system concepts that define the general set of activations that describe the model. The Virtual manufacturing activities (A0) encompasses our entire system. The client and the machine are the two external players interacting with the system. The client puts the demand on the system, and the machines are instructed to satisfy those demands. This VM activity is accomplished using a Data System, that is, a computer. All the different requests are received from the client. They are acted upon accordingly and the corresponding response is sent back to the user. Machine control signals are sent to the machine, and the resultant feedback is received.

The next level of decomposition is shown in Figure 2b. The VM activities can be broadly classified into the web server activities (A1) and the manipulation/retrieval of data (A2). The web server activity is implemented using the Java Virtual Machine (JVM) and the latter using a database. Both of these implementations are part of the data system. The design constraints and machine specification affect the performance of both the activities. The web server activity interacts with the database to update data or request information and receives the consequent response.

The decomposition of the web server activities is shown in Figure 3. It consists of five distinct activities – Design the part, Plan the process, Generate code, Simulate the machining operation and Control the machine. All the five activities are carried out by their own dedicated module written in Java TM and by using database connectivity. The client sends in a request to modify design, which is handled by the ‘Design the part’ activity (A11). The changes in the design are controlled by the design constraints and machine specifications. The necessary changes are made in the design, the new design specification is updated in the database, and the feedback is sent to the user.

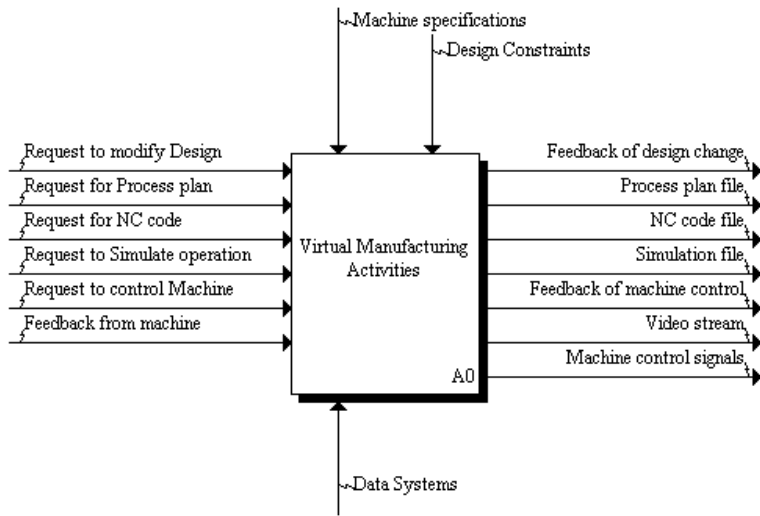


Figure 2(a). Top level or A0 diagram of the VM System

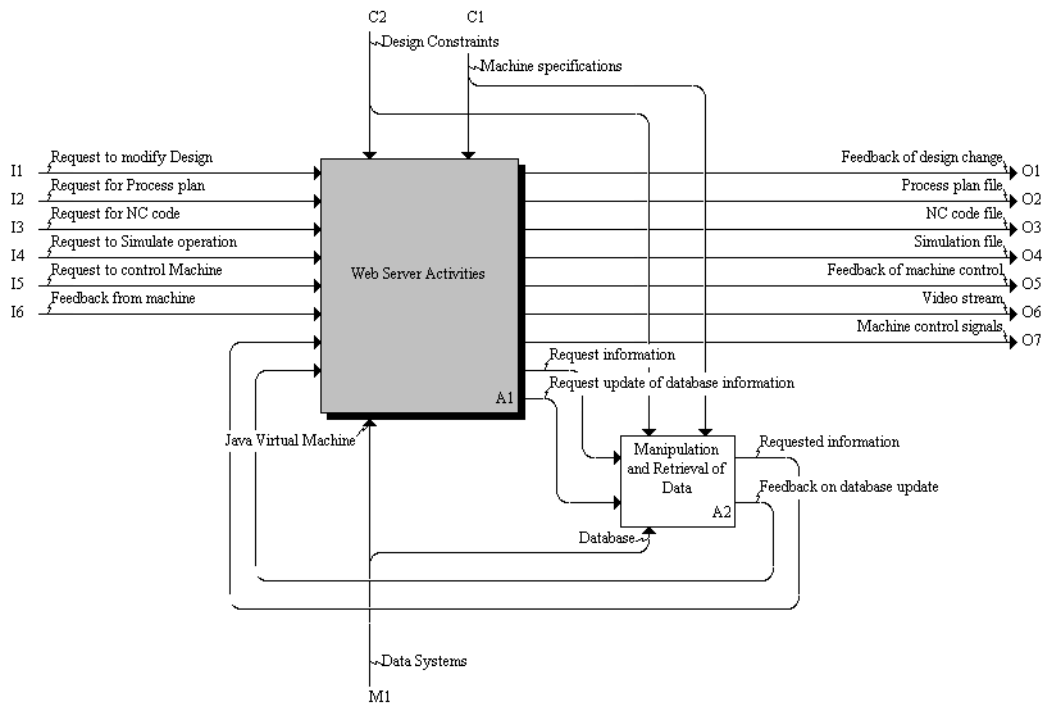


Figure 2(b). Second level decomposition of the VM System

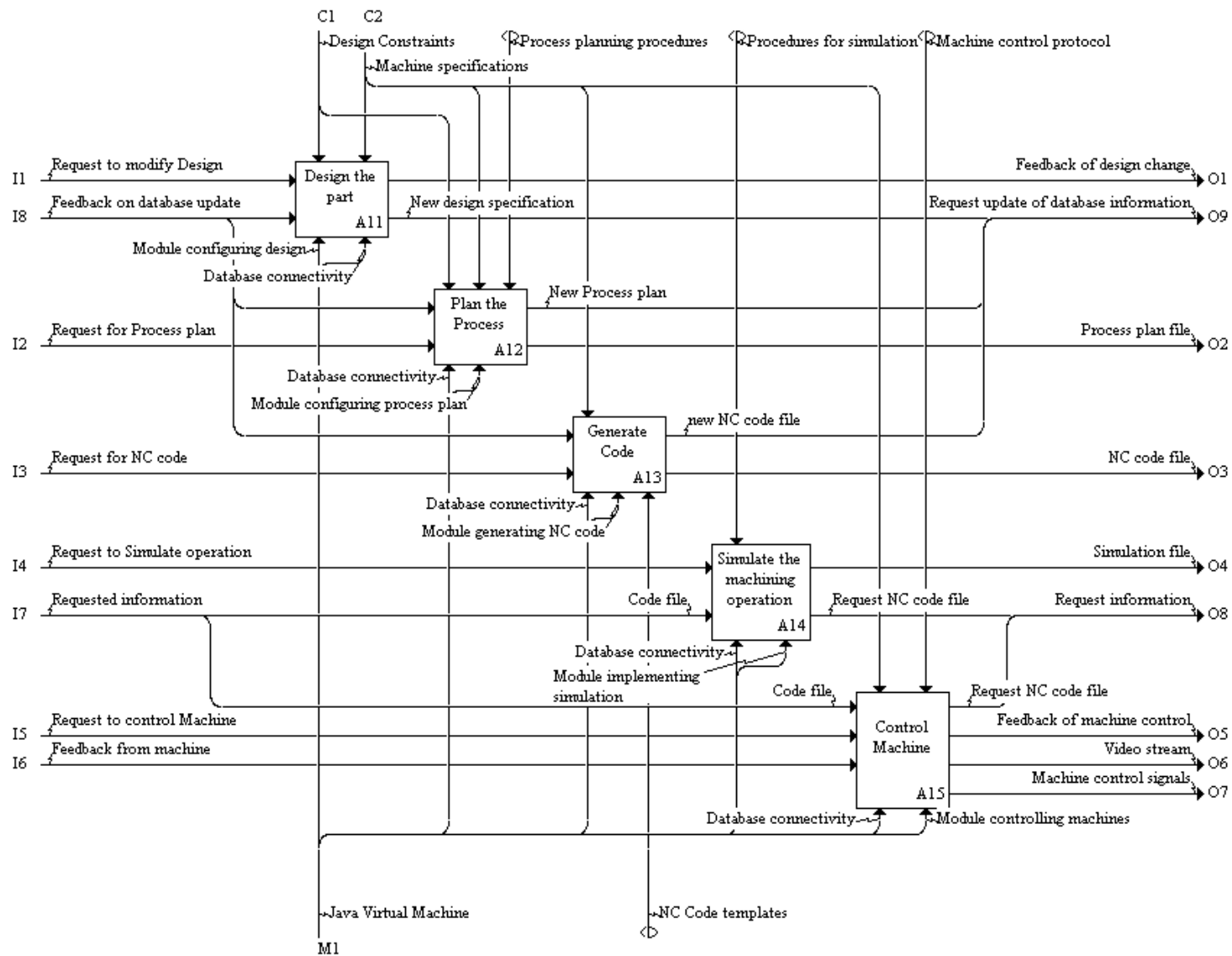


Figure 3. Third level decomposition of the VM System

The client's request for the process plan is handled by the 'Plan the process' activity (A12). A new process plan is prepared and a copy of the file is sent to the database and another copy goes to the client. The Process planning procedure laid down by the organization also controls this activity. The request for NC code from the client is handled by the 'Generate code' activity (A13). This activity requires the additional use of NC code templates. The new NC code file is updated in the database and a copy of the same is sent to the client. When the 'Simulate the machine operation' activity (A14) receives the request to simulate the design, it retrieves the NC code file corresponding to the client's design choices from the database. Using the NC code file and the simulation procedures, it generates the simulation and sends it to the client. The final request handled by the server is the request to control the machine. It is taken care of by the activity 'Control machine' (A15). Upon receiving the request, control signals are sent to the machine, and the live video stream is buffered back to the user. Upon completing the task, the machine sends the feedback to the activity. On getting the feedback, the machine terminates the live video feed and sends an appropriate response to the client.

The modeling can be decomposed into further levels until every minute activity is recorded. But the purpose here is to provide an overall understanding of the system's activities, which is achieved in these three levels of abstraction. The next section deals with the implementation of the system.

6. IMPLEMENTATION OF THE SYSTEM

The proposed system has been developed at the Computer Integrated Manufacturing Lab at The University of Arizona with specific use of a Machining Center to implement the model.

The database for the system is maintained using Microsoft Access TM. The information pertinent to the working of the system is as shown in Table 2.

Table 2. Data available in the database (not exhaustive)

Data about Work piece	Material, dimensions, hardness, etc.
Data about machines & equipment	Machine name, description, class, tool load/unloading time, part loading/unloading time, etc.
Data about tools	Tool name, type, diameter, material, etc.
Data about costs	Machine cost, set-up cost, cutter cost, etc.
Data for design	Dimensional and Geometric tolerance required, surface finish etc.
Data about clients	Client name, user ID, password, time system accessed, number of access, designs created, designs machined etc.
Data for process planning	The speed and feed parameters according to the operation performed, material & diameter of tool used, material & hardness of work piece. †

† - Obtained from the Machining Data Handbook

In the database, temporary tables are created to store the custom-made design details and process plan that are deleted after the client terminates his connection.

Communication between the client and the server is managed using the HTTP protocol. The communication typically takes place over a TCP/IP connection on the Internet. The system has a single server that the user accesses through the Internet. The default capability of the web browser to upload new web pages is used. Java Web Server™, a product of Sun Microsystems, is used to host the web pages and servlets. It is a highly flexible, secure, and platform independent tool satisfying our requirements for the implementation of the system (Sun Microsystems's Java Web Server *website*). The users send their requests along with the servlet name that will handle the request. The corresponding servlet is then loaded onto the server's JVM. A separate thread within the web server process handles each user's request.

One or more Java™ servlets together act as a single module. The five different modules, one for each activity, are shown in Figure 3. These modules retrieve or update the data in the database using JDBC. Each of these modules carries out its own error checking and feedback algorithms. Only the final results are exchanged among the modules. The required mutual exclusion is done to ensure that only one user can access the machines in the machine control module. The modules are autonomous. This facilitates future modifications and provides scalability without affecting the system as a whole.

The design module checks the validity of the user's design choices using the design protocol and updates the client design in the database. Based on the current facilities available the design choices are restricted, which is explained in Section 7 - Demonstration of our System. The process planning module use the design data to create a new process plan. The corresponding speed & feed parameters data, and cost data are queried from the Access™ database. (Refer 'Data for Process planning' and 'Data about costs' in Table 2). The machining time (Tm), material-handling (Th) time, total production time (Tpr) and the total production cost (Cpr) is calculated. A new temporary table, as shown in Figure 4, is created in the database to store the process plan.

NC code generation modules use NC code template files to create new custom NC files. NC code files for all of the letters in the alphabet are stored in the server. When the custom created design is received, then the X, Y, Z co-ordinates in the NC code are changed according to the requirements.

Op Code	Description	Machine ID	Speed	Feed	Tool ID	Tm	Th	Tpr	Cpr	Notes
▶	1 Clamp part to Machine bed to drill	Automatic Clamping	0	0		0	0.2	0.2	0.1166666667	
	2 Drill character 1	M1	100	0.012	T2	0.0872638889	1.25	1.3376	0.20360522	
	3 Drill character 2	M1	100	0.012	T2	0.0727199074	1.17	1.2427	0.1960903	
	4 Drill character 3	M1	100	0.012	T2	0.0855246875	1.35	1.4355	0.1627316	
	5 Unclamp part from Machine bed	Automatic Clamping	0	0		0	0.2	0.2	0.1166666667	
*	0		0	0		0	0	0	0	

Figure 4. Database table showing the process plan

The Simulation module performs a simulation of the tool movement. The custom created NC code file is used for this purpose. First, a Java™ applet is downloaded to the client's machine. It then establishes connection with the servlet implementing the simulation module. This servlet reads the custom created NC code file, interprets it and sends the coordinates to the applet. The applet performs the plotting of the points and traces the tool route on the users screen.

The machine control module is the crux of our model. A C program, involving parallel port programming, implements the machine control logic. The program is invoked from the machine control module. The C program gets as input the NC code file. The program interprets the file, converts the text into bit signals, and sends it to the appropriate axis control stepper motor to move the axis the specified distance. Upon successful completion of machining, the C program gives feedback to the machine control module. Meanwhile, the web camera is invoked from the servlet. A live channel is created, similar to the one used in audio and video live streaming data, to stream live video feed to the user.

The client side interfaces are created using HTML. The site is easily navigable and user friendly. For running the simulation, a Java TM applet with the communications with the server is downloaded to the client machine.

Multiple servers can be used to improve the efficiency and power of the system. As the server is programmed in Java TM it is easy to replicate. Similarly the program controlling the actual machines can also be distributed to different machines.

7. DEMONSTRATION OF OUR SYSTEM

This section presents a demonstration of our system to provide a better insight of its capabilities.

The user on the client connects to the server by typing in the URL address in the web browser. The user will then see a login page, which prompts the user to enter the username and password. This form-based custom authorization provides basic security for the system. The username and password are handled by a dedicated servlet. The servlet, after verifying the validity of the client, creates a persistent cookie and sends it to the user. A cookie is a bit of information sent by the web server to a browser that can later be read back by the server. When the client browser receives the cookie, it saves the cookie and thereafter sends the cookie back to the server each time it accesses a page on the server. The cookie is used for session tracking. Once the user terminates his connection with the server, the cookie is automatically deleted. Each time the client sends a request, the server checks the cookie sent by the user's browser. If it is invalid, then the client is redirected to the login page.

The user, after successful login, is presented with the main menu. The menu provides the options to modify the design, view the process plan, NC codes, and simulation, and control the machine remotely. On selecting the design link, the user is presented with the page as shown in Figure 5. Current design features allow the user to select a maximum of 3 letters to be machined, the material of the work piece, the dimension of the characters, the depth of cut, and the tool type. Three types of tools are available based on their diameter. The above features provide enough variability to make the system dynamic. More features can be added to the current system at the cost of increasing system complexity but without significantly altering the architecture. Also, as the system is structured in a modular fashion, the complexity needs to be increased only in the design module. Refer to Section 8 – Applications and future research, to find future extensions on this module. The user after selecting the appropriate choices clicks the submit button. This action invokes the servlet handling the design. This servlet receives the user's chosen design options. The servlet validates the choices and then stores the information in the Microsoft Access TM database using JDBC. The information stored in the database includes details about the client along with the design choices made.

The next step is to view the process plan for making the corresponding design. The servlet computing the process plan uses the information about the material and machine along with the

client's preferences. A new table dedicated for the client is created where all the data is stored. The process plan generated is as shown in Figure 4. The table is later deleted when the user logs off or terminates his connection. The data is retrieved from the database and buffered to the user. The important information such as the time taken, cost, etc are highlighted for better usability. The entire process plan is also made available to the user.

The Code generation feature allows the user to view the NC code required to machine the product. A copy of the NC code template files is made. These codes are then modified according to the user specifications to get the desired design.

The final step before actual machining is to view the machining operation using simulation. It provides a means of confirming the process and also a visual interface for the user to ensure that they get what they want. If the user is unsatisfied with the design, they can modify the design to suit their needs and tastes.

Controlling the machine is the major focus of this system. Using this feature the user can actually control the machine. The servlet in charge of controlling the machine is invoked. It in turn calls the machine specific software written in C to control the machine. The user created NC code file is provided as input to the C program. The machine control program uses parallel port programming to invoke the Machining center. The corresponding design is traced on the work piece using the desired tool. The entire machining process is made available live to the user via the web camera.

Design Selection

Please choose a design you would like to machine using the drop-down box below. The default is "SIE". To exclude a character select the blank from the drop-down box.

Workpiece material Aluminium ▼

Character 1	Character 2	Character 3
S ▼	I ▼	E ▼

Character dimensions

Length is inches (ranges from 1" to 2.5")

Width is inches (ranges from 1" to 2.5")

Height is inches (ranges from 0.1" to 0.5")

Thickness of line is inches (1/8", 1/4" or 3/8")






Figure 5. Design selection page as seen by user

8. APPLICATIONS AND FUTURE RESEARCH

The potential benefits of this system in both academic and industrial areas are numerous.

As the system itself has been made of standalone modules, each can be expanded to make them more generic and global in scale. The constraints of the existing system can be relaxed to provide more flexibility from the users viewpoint. The design module can be enhanced so that more varieties of parts can be manufactured. CAD files can be uploaded by the users, which will be interpreted by the server and stored in the database. Features can be included to get the user written NC code and check it using simulation. The simulation itself can be modified using better 3D graphics. The scope of the simulations can be extended from applications for a single machine to advanced versions that show interactions among different machines within an automated cell and between different such cells in a shop floor. Different machines of varying capacities, robots and buffers can be added to make a completely automated cell that can be remotely controlled. Scheduling and optimization techniques can also be added. Current work is going on in these directions. Other features can be added to make the system much more effective and efficient.

Wide access to fixed resources would help accelerate work and facilitate collaboration that reaches over national and international boundaries. Convenient access to innovative machines can thus be provided. Companies are trying to achieve higher customer satisfaction by providing specially designed products, tuned to customer desires. But this increases the cost of production which adversely affects the company's competitive ability. But with the advent of the Internet, customization on a bigger scale is possible. Mass customization of products leading to an enormous number of small-individualized orders can, in fact, give rise to a new commerce.

Distance learning is becoming more and more popular. Distance education, especially for lab courses, can be realized without any compromise. Students and teachers across geographical regions can collaborate effectively for their mutual benefits.

9. CONCLUSION

This paper presented an Internet based manufacturing system. This system offers a tremendous opportunity for a new kind of communication. The implementation of such a system has been made possible only by the advent of the Internet and distributed programming, increasing the flexibility of the client-server-based approach. The architecture, detailed functional models, and implementation of the system have been presented. As part of the architecture, the concept of an application independent server composed of autonomous modules is proposed. Ideally the user can access the system from any part of the world through the Internet.

10. REFERENCES

1. Shukla, C., Vazquez, M., and Frank Chen, F., 1996, "Virtual Manufacturing: an Overview", *Computers and Industrial Engineering*, 31, 79-82.
2. Di Stefano, A., Fazzino, A., Lo Bello, L., and Mirabella, O., 1997, "Virtual Lab: A Java Application for Distance Learning", *IEEE Emerging Technologies and Factory Automation Proceedings*, 95-98
3. Layne, S.P., and Beugelsdijk, T. J., 1998, "Mass customized testing and manufacturing via the Internet", *Robotics and Computer-Integrated Manufacturing*, 14, 377-387

4. Küchlin, W., Speck, A., Lumpp, T., and Gruhler, G., 1997, "Highrobot: Telerobotics in the Internet", *IEEE Emerging Technologies and Factory Automation Proceedings*, 115-120
5. Singh, N., 1996, *Systems Approach to Computer Integrated Manufacturing*, John Wiley & Sons.
6. Lawrence Associates Inc., 1994, *Virtual Manufacturing User Workshop Technical Report*.
7. Dessouky, M.M., Verma, S., Bailey, D.E., and Rickel, J., 2001, A methodology for developing a web-based factory simulator for manufacturing education. *IIE Transactions*, 33, 167-180
8. Dessouky, Y.M., Roberts, C.A., and Beaumarriage, T.G., 1995, Object-oriented simulation architecture with real time capabilities. *International Journal of Production Research*, 33(9), 2471-2492.
9. Adamczyk, Z., Kociopek, K., 2001, CAD/CAM Technological environment creation as an interactive application on the Web, *Journal of Materials Processing Technology*, 109, 222-228.
10. Lee, Jay, 1998, Teleservice engineering in manufacturing: challenges and opportunities. *International Journal of Machine Tools and Manufacture*, 38, 901-908.
11. Chang, T., Wysk, R.A., and Wang, H., 1998, *Computer-Aided Manufacturing*, 2nd ed., Prentice-Hall.
12. Hunter, J., 1998, *Java Servlet Programming*, 1st ed., O'Reilly & Associates Inc.
13. Molva, R., 1999, Internet security architecture, *Computer Networks*, 31, 787-804.
14. Mayer, R. J., 1992, *IDEFØ Function Modeling Method Report*. Knowledge Based Systems Inc. College Station, TX.
15. Shin, J., Joo, J., Choi, I., Han, S.H., and Cho, H, 2000, "A prototype of Virtual Reality System through IDEF Modeling for product configuration and analysis", *International Journal of Industrial Engineering*, 7(1), 15-25.
16. Sun World website
(<http://www.sunworld.com/sunworldonline/swol-12-1998/swol-12-servlets.html>)
17. Sun Microsystems web site
(<http://java.sun.com>)
18. Sun Microsystems's Java Web Server website
(<http://www.sun.com/software/jwebserver/index.html>)