2004

A data exchange system in e-manufacturing.

Xinhui Zhong
University of Windsor

A DATA EXCHANGE SYSTEM IN E-MANUFACTURING

By

Xinhui Zhong

A Thesis

Submitted to the Faculty of Graduate Studies and Research

Through the Department of Industrial and Manufacturing System Engineering

In Partial Fulfillment of the Requirement for

The Degree of Master of Applied Science at the

University of Windsor

Windsor, Ontario, Canada

2003

Your file Votre référence
ISBN: 0-612-92505-6
Our file Notre référence
ISBN: 0-612-92505-6

ABSTRACT

The emergence of the Internet has fundamentally changed the way that people communicate and view the world. As a new manufacturing paradigm, e-Manufacturing is about using the web-enabled and tether-free infotronic technologies for manufacturing operations. Although e-Manufacturing has already been an often-mentioned topic, in the past decade, practical implementation has been slow to develop due to insufficient technologies to handle information flows connected with e-Manufacturing.

Recently, there is considerable interest in the area of Internet enabled distributed systems. Examples of these works include online part measurement [Grimaldi, 1998] and Distributed Rapid Prototyping Via the Internet [Tay, 1999]. The research target of these work focus on remote manufacturing control and monitoring via the Internet. However, Data exchange, an important part for global co-operation, hasn't been fully studied and there is not a lot of work that has been done in previous research. In this thesis, efforts have been made to highlight the role of data exchange in Internet-enabled manufacturing, and, an Internet-based Data Exchange System has been developed with JSP and Oracle database. The developed system has advantage in commonality and capability of data-transaction over the previous work.

As an interesting complement to the study of previous researches, a novel methodology is also proposed for utilization of remote resource via the Internet, using commercial software 'PC Remote Access'. The implementation of this methodology has successfully been done to use software including AutoCAD, MasterCAM and Catalyst over the Internet. The biggest problem for the application of this approach lies in the fact that 'PC remote Access' software cannot make one PC get access to another PC which is behind firewall. However the software supplier has announced that the problem will be addressed in the near future.

ACKNOWLEDGEMENTS

I am deeply indebted to my supervisor, Dr. Waguih H. ElMaraghy for the guidance and the support he had extended to me throughout the course of my M.A.Sc. Program. I would like to express my deepest gratitude to my supervisory committee, Dr. David Ting, Dr. G. Zhang, for their comments and time in reviewing my thesis.

I would like to thank the Industrial and Manufacturing Systems Engineering Program secretary, Ms. Jacquie Mummery, and department technical support, Mr. Ram Barakat. I would like to thank IMSE Research/Administrative Assistant, Mrs. Monique Gagnon and Mrs. Zaina Batal for their help in proofreading my thesis.

I would like to extend my thanks to my friends and colleagues who helped me during my stay in the Intelligent Manufacturing Systems Center, namely, Mrs. Jill Urbanic, Mrs. Omayma Nada, Mr. Dave Faccend, Mr. Ayman Youssef, Mr. Amr Shabaka, Mr. Ahmed Deif, Mr Anas Kazaal and Mr. Onur Kuzgunkaya.

I would like to acknowledge the great help and insight that I received from my dear friends, Mr. Deven Parekh, Mr. Xudong Xie, Mrs. Amber Zhang, and Mr. Tim Huang. The priceless discussions I had with them were a great help to accomplish the work presented in this thesis.

I would not have accomplished this work without the support, help, love and insight of my great parents, Mr. Huarong Zhong, and Mrs Yumei Liu, my brother Jerry, and my sister Joan.

Last, but not the least, the love and faith of my dear wife, Angel, have always been my inspiration. I would like to present this work to her, and our expected baby.

NOMENCLATURE

AES	Advanced Encryption Standard
B2B	Business-to-Business distribution channel functionality
CCD	Charge-Coupled Device Camera
CNC	Computer Numeric Control
CRM	Customer Relation Management
EAI	External Authoring Interface
EDI	Electronic Data Interchange
EPM	Enterprise Production Management
ERP	Enterprise Resource Planning
F2B	Factory-to-Business distribution channel functionality
FDM	Fused Deposition Modeling, one of the RP techniques
JDK	Java Development KIT
JSP	Java Server Pages
LOM	Laminate Object Manufacturing, one of the RP techniques
MES	Manufacturing Execution Systems
MRP	Manufacturing Resource Planning
RDBMS	Oracle Relational Database Management System
RP	Rapid Prototyping
SCM	Supply Chain Management
SFM.	Surface Feet Per Minute
SGC	Solid Ground Curing, one of the RP techniques
SLA	Stereolithography, one of the RP techniques

SLS	Selective Laser Sintering, one of the RP techniques
SML	Catalyst Modeling Language
SQL	Structured Query Language
STL	Standard Triangulation Language
TLO	Tool Length Offset
TMF	Tele-Manufacturing Facility
VE	Virtual Environment
VNC	Virtual Network Computing Software
VR	Virtual Reality
VRML	Virtual Reality Modeling Language
XML	Extensible Markup Language

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

CHAPTER 1 – INTRODUCTION

This chapter presents the basic concepts that would help readers understand the further work. Section 1.1 reviews e-Manufacturing in manufacturing industry. Section 1.2 introduces two typical processing, Rapid Prototyping and CNC Machining, the previous researches on Internet-enabled manufacturing had most of their implements in the Rapid Prototyping, and the implements of this thesis are related to both processing. Section 1.3 presents an overview of this thesis.

1.1 General Introduction to e-Manufacturing

'Briefly speaking, e-Manufacturing is a transformation system that enables the manufacturing operations to achieve predictive near-zero-downtime as well as to synchronize with the business systems through the use of web-enabled and tether-free (i.e. wireless, web, etc.) infotronics technologies'[Lee, 2001].

Roach [2001] emphasized its importance in data-exchange. "With e-Manufacturing, information is shared freely between business and plant floor. Allowing organizations to better plan, execute, and deliver to meet customer needs [and so it] ties systems together and enables real-time decision making across the enterprise". Recently, e-Manufacturing is an effective method for communication and for sharing information related to manufacturing throughout the entire enterprise and supply chain via the Internet.

1.1.1 The worldwide development of Internet

Almost all studies agree that e-Manufacturing owes its advent to the feasibility of Internet applications. First came e-mail, then e-commerce, and then e-business. In each case, the 'e' indicated taking an existing process and adding to it the Internet's quick and easy information exchange. This simple definition can also be applied to e-Manufacturing. But adding the Internet's speed, and collaboration capabilities to existing

manufacturing processes will radically reshape it, demanding that manufacturing plants master new capabilities, and that engineers develop new priorities.

The recent researches on Internet development revealed the evolution trend [Olmos and López, 2000]. The following figures have shown the great impact of Internet:

1. Adopted at the tremendous speed after the year 1992;
2. Functionality gradually switched from Publication to Processes;

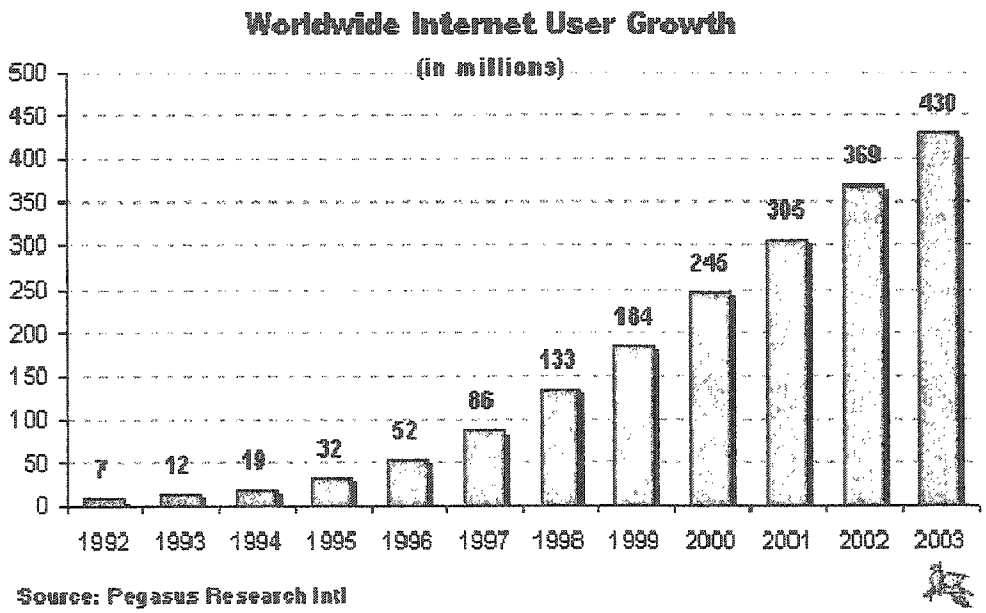

Figure 1 The growth of Internet Users [Emilio's Website]

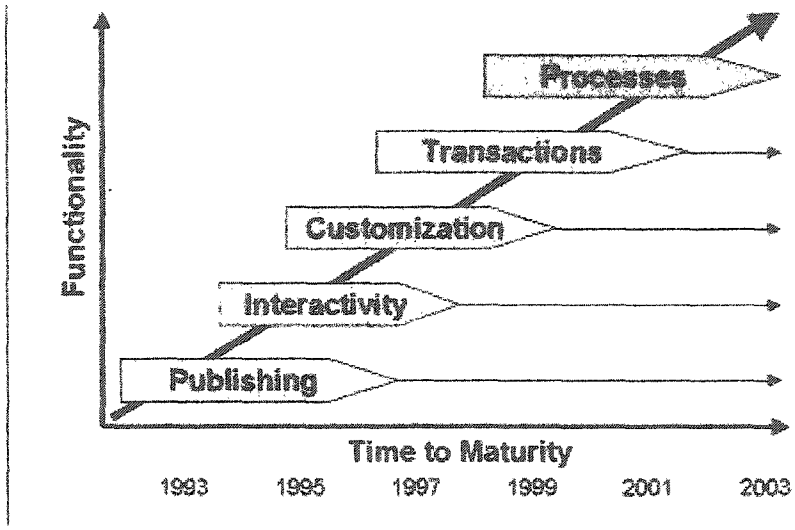

Figure 2 Evolution of Internet Functionality [USC Website, 2000]

1.1.2 Development of e-Manufacturing

In 1993 Lynch and Rose summarized the important dates of Internet development. Recent studies show that e-Manufactuing had a synchronous development in the last 2 decades. Table 1 below demonstrates the close timely relation of the two techniques.

Table 1 Internet and e-Manufacturing Development

Year	Internet	e-Manufacturing
Early 1960s	First concept & Role Model formation	
1968	Lab Experiments	
1973-1974	TCP/IP was designed & The first applications	
1980s		First concept & Experiments

1995	Netscape available	
1990s		First e-Manufacturing Applications, mainly in heavy industry
1997		'Tech Revolution' of e-Manufacturing

In 2000, L.A. Gordon studied e-Manufacturing's origination, in his report '*The industrial Internet: a total digital enterprise mindset*', he emphasized the huge impact of the fundamental shifts in technology initiated by 'Tech Revolution'. In that paper he also pointed out that "The demarcation year, 1997, seems to be the pivotal point in this migration".

According to L.A. Gordon's research, Table 2 below summarizes the changes in the year 1997 that Internet had brought for e-Manufacturing development:

Table 2 – The Internet technologies' Major Impact on e-Manufacturing in 1997 [Gordon, 2000]

	Before 1997	**After 1997**
CNC Control	Hardware-driven CNC	Software CNC (SCNC)
Industrial Ethernet	Local Area Networks (LANs)	Industrial Ethernet
PC-Based C ontrols	Established PLC world	'Open Systems' & Trans-enterprise networking
Industrial Internet	Manufacturing Resource Planning (MRP)	Internet-based EDI & ERP system
Distribution Chains	Static Distribution Networks	Dynamic Internet-enabled
Business-to-Business	Pure Network-based Commerce	Emergence of F2B initials

1.1.3 System Architecture of e-Manufacturing

The figure below illustrates an e-Manufacturing architecture that comprises of the following elements [Lee, 2002]:

- Data Gathering
- Data transformation
- Prediction
- Optimization
- Synchronization

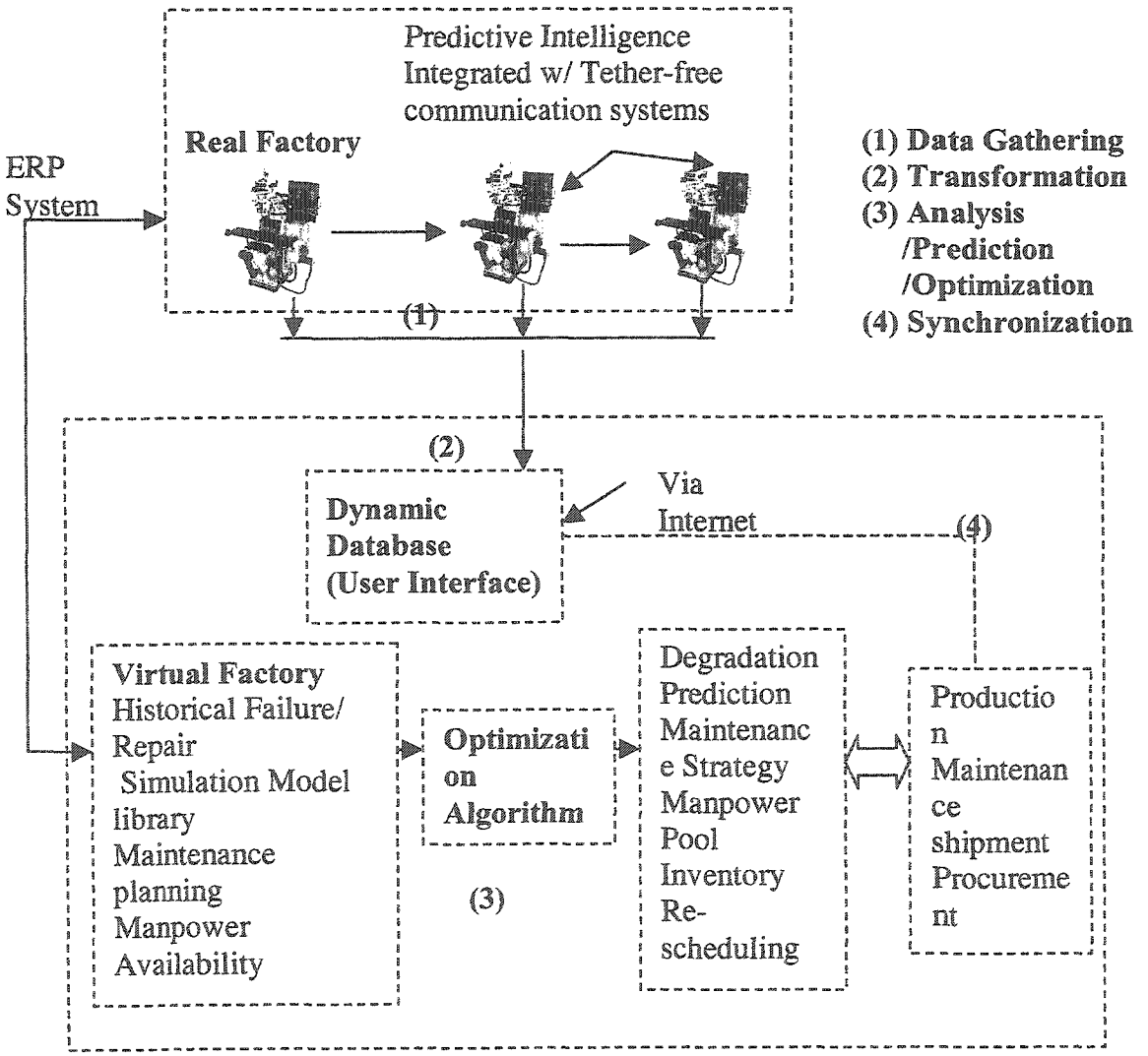

Figure 3 An e-Manufacturing Architecture [Lee, 2002]

1.1.4 Application Areas of e-Manufacturing

E-Manufacturing as a new generation of product development solution allows manufacturers all over the world to speed up and slim down everything from design and procurement to manufacturing and logistics. It has been employed in a wide range of manufacturing:

- Product design
- Prototyping
- Factory Layout Design and Visualization
- Assembly process planning and Simulation

1.2 Introduction to CNC Machining and Rapid Prototyping

The concept of 'Internet Manufacturing' or 'Distributed Manufacturing' has long been widely used in the manufacturing and design industries. One implementation has already been done on Rapid Prototyping processing [Tay, 1999]. In this thesis, efforts will be made to create a Java-based data-exchange system, for the global cooperation via Internet. And as an interesting complementation to the previous studies, a novel methodology will also be proposed for Internet Manufacturing remote control.

In order to understand the further chapters, here in this section, we have a brief introduction of the mentioned processing- Rapid Prototyping and Computer Numeric Control (CNC).

1.2.1 Rapid Prototyping

1.2.1.1 Definition of RP processing

Rapid Prototyping (RP) is a technology that takes a three-dimensional computer model and builds up a plastic-like part by "printing" layers of material one on top of another. Layer by layer, RP machines fabricate plastic, wood, ceramic and metal objects

using thin horizontal cross sections directly from a computer generated model. Its speed and low cost allow design teams to confirm their new designs early and frequently in the process.

1.2.1.2 RP vs Conventional Methods

Unlike milling machines, which are subtractive in nature, RP systems are additive processes that join together liquid, powder or sheet materials to form the parts. It is true that rapid prototyping (notice the lowercase) can be achieved using conventional methods such as NC milling and hand carving. However, the term RP is normally reserved for the new technologies that build parts by adding material instead of removing it.

1.2.1.3 Applications for Rapid Prototyping (RP)

Rapid Prototyping is good for the manufacturing parts that have the following features:

1) Single unique item or small number of copies needed;

2) Shape of object is in computer form;

3) Shape is too complex to be generated in any other way.

1.2.1.4 Benefits of using Rapid Prototyping (RP)

Rapid Prototyping (RP) offers advantages in many applications compared to classical subtractive fabrication methods such as milling or turning:

- Objects can be formed with any geometric complexity or intricacy without the need for elaborate machine setup or final assembly;
- Rapid prototyping systems reduce the construction of complex objects to a manageable, straightforward, and relatively fast process.

It has been claimed that Rapid Prototyping (RP) can reduce new product cost by up to 70% and the time to market by 90% [Phsm, 1998].

1.2.1.5 Technologies for Rapid Prototyping (RP)

There are many rapid prototyping techniques that are commercially available. The major rapid prototyping systems include Stereolithography (SLA), Solid Ground Curing (SGC), Selective Laser Sintering (SLS), Laminate Object Manufacturing (LOM), Fused Deposition Modeling (FDM), and 3D Plotting. Although rapid prototyping machines function similarly, the specific technologies they use widely differ.

The specific methods are introduced as below:

(a) Stereolithography (SLA).

Stereolithography builds plastic parts or objects a layer at a time by tracing a laser beam on the surface of a vat of liquid photopolymer. This class of materials, originally developed for the printing and packaging industries, quickly solidifies wherever the laser beam strikes the surface of the liquid. Once one layer is completely traced, it's lowered a small distance into the vat and a second layer is traced right on top of the first. The self-adhesive property of the material causes the layers to bond to one another and eventually form a complete, three-dimensional object after many such layers are formed.

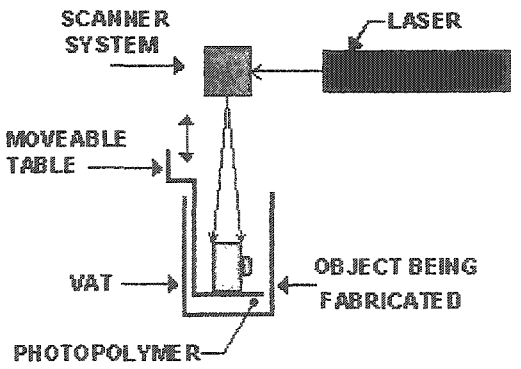

Figure 4 Illustration of Stereolithography (SLA)

(b) Fused Deposition Modeling (FDM).

A plastic filament is unwound from a coil and supplies material to an extrusion nozzle. The nozzle is heated to melt the plastic and has a mechanism which allows the flow of the melted plastic to be turned on and off. The nozzle is mounted to a mechanical stage which can be moved in both horizontal and vertical directions. As the nozzle is moved over the table in the required geometry, it deposits a thin bead of extruded plastic to form each layer. The plastic hardens immediately after being squirted from the nozzle and bonds to the layer below. The entire system is contained within a chamber which is held at a temperature just below the melting point of the plastic. Several materials are available for the process including ABS and investment casting wax.

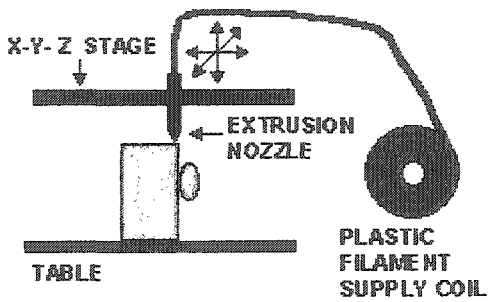

Figure 5 Illustration of Fused Deposition Modeling (FDM).

(c) Inkjet based systems.

This method uses a single jet each for a plastic build material and a wax-like support material, which are held in a melted liquid state in reservoirs. The liquids are fed to individual jetting heads which squirt tiny droplets of the materials as they are moved in X-Y fashion in the required pattern to form a layer of the object. The materials harden by rapidly dropping in temperature as they are deposited. After an entire layer of the object is formed by jetting, a milling head is passed over the layer to make it a uniform thickness. Particles are vacuumed away as the milling head cuts and are captured in a

filter. The process is repeated to form the entire object. After the object is completed, the wax support material is either melted or dissolved .

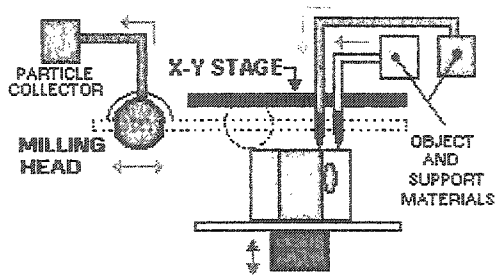

Figure 6 Illustration of Inkjet based systems.

(d) Three Dimensional Printing (3DP).

The process starts by depositing a layer of powder object material at the top of a fabrication chamber. To accomplish this, a measured quantity of powder is first dispensed from a similar supply chamber by moving a piston upward incrementally. The roller then distributes and compresses the powder at the top of the fabrication chamber. The multi-channel jetting head subsequently deposits a liquid adhesive in a two dimensional pattern onto the layer of the powder which becomes bonded in the areas where the adhesive is deposited, to form a layer of the object.

Once a layer is completed, the fabrication piston moves down by the thickness of a layer, and the process is repeated until the entire object is formed within the powder bed. After completion, the object is elevated and the extra powder brushed away leaving a "green" object.

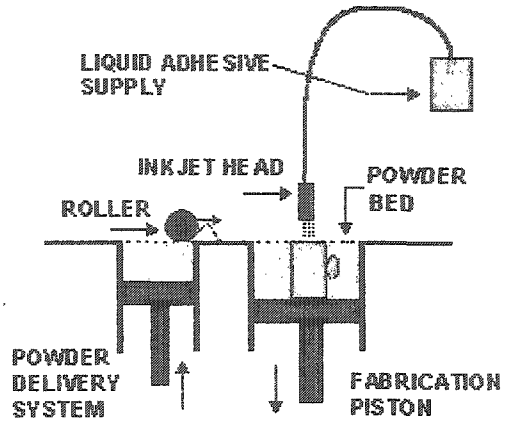

Figure 7 Illustration of Three Dimensional Printing (3DP).

(e) Selective Laser Sintering (SLS).

Thermoplastic powder is spread by a roller over the surface of a build cylinder. The piston in the cylinder moves down one object layer thickness to accommodate the new layer of powder. The powder delivery system is similar in function to the build cylinder. Here, a piston moves upward incrementally to supply a measured quantity of powder for each layer.

A laser beam is then traced over the surface of this tightly compacted powder to selectively melt and bond it to form a layer of the object. The fabrication chamber is maintained at a temperature just below the melting point of the powder so that heat from the laser need only elevate the temperature slightly to cause sintering. This greatly speeds up the process. The process is repeated until the entire object is fabricated. After the object is fully formed, the piston is raised to elevate it. Excess powder is simply brushed away and final manual finishing may be carried out.

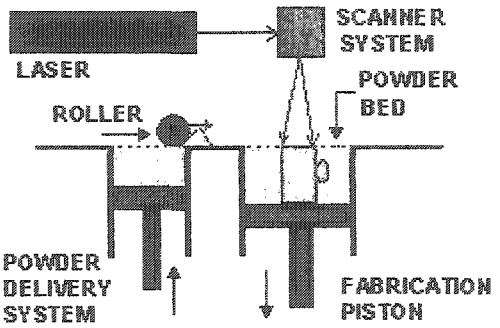

Figure 8 Illustration of Selective Laser Sintering (SLS).

(f) Laminated Object Manufacturing (LOM).

Profiles of object cross sections are cut from paper or other web material using a laser. The paper is unwound from a feed roll onto the stack and first bonded to the previous layer using a heated roller which melts a plastic coating on the bottom side of the paper. The profiles are then traced by an optics system that is mounted to an X-Y stage.

After cutting of the layer is complete, excess paper is cut away to separate the layer from the web. Waste paper is wound on a take-up roll.

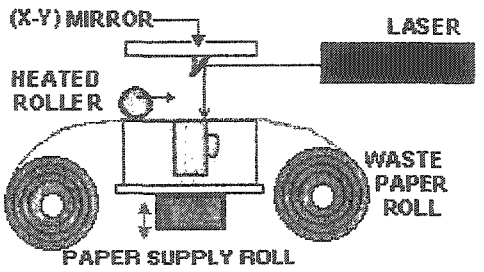

Figure 9 Illustration of Laminated Object Manufacturing (LOM).

(g) Laser Engineered Net Shaping (LENS)

Laser Engineered Net Shaping (LENS) is a technology that is gaining in importance and in early stages of commercialization. A high power laser is used to melt metal powder supplied coaxially to the focus of the laser beam through a deposition head. The laser beam typically travels through the center of the head and is focused to a small spot by one or more lenses. The X-Y table is moved in raster fashion to fabricate each layer of the object. The head is moved up vertically as each layer is completed. Metal powders are delivered and distributed around the circumference of the head either by gravity, or by using a pressurized carrier gas. An inert shroud gas is often used to shield the melt pool from atmospheric oxygen for better control of properties, and to promote layer-to-layer adhesion by providing better surface wetting.

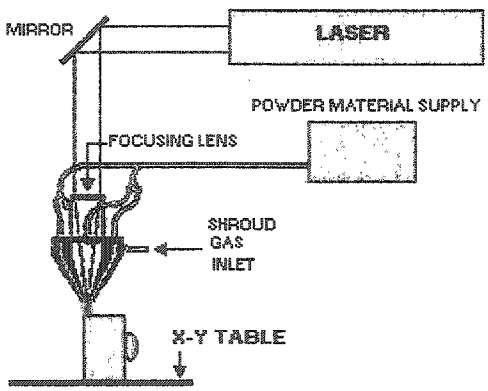

Figure 10 Illustration of Laser Engineered Net Shaping (LENS).

1.2.2 Overall Introduction to Computer Numeric Control (CNC)

1.2.2.1 Definition

Computer Numerical Control (CNC) refers to the control of a machine tool using numbers and letters. It is a system in which programmed numerical values are directly inserted and stored on some form of input medium, and automatically read and decoded to cause a corresponding movement in the machine which it is controlling.

1.2.2.2 Cutting Tools

CNC Cutting tool include milling machine boring bar, Lathe tools, End mills and Drill, Ream, Tap, and Grinding wheel and bits.

1.2.2.3 Tool Length Offset (TLO)

The Tool Length Offset value is the distance from the tip of the tool with the spindle in the home position, to the Z0 plane. TLOs must be set for each tool used in the current job. They can be set using a height gage, reference tool, fixture location, or on the stock to be machined. Once the values are determined, they are stored in the controller for use during program operation.

A more efficient way to set TLOs is to measure from the spindle to each tool tip. This is better method because the values need only be set once regardless of each different type and size of part being manufactured. Of course, tool breakage or replacement requires a reset of the individual TLO when using either method.

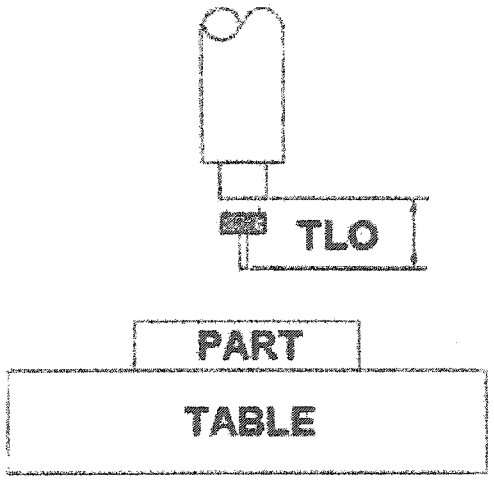

Figure 11 Tool Length Offset Illustration

1.2.2.4 Machine Setup and Operation

The basic CNC operation include the following:

1. Startup - Usually there is a main power switch or circuit breaker. Some machines also require air pressure or hydraulics.
2. Home Position
3. Tool Length Offsets
4. Setting up a Part
5. Origin
6. Program Loading
7. Viewing Program

1.2.2.5 Machining Data

Here are some of the most common terms used for expressing cutting data:

Spindle speed - Spindle speed is the rotational speed of the spindle and tooling. This value is usually expressed in RPM. (Revolutions Per Minute)

Feedrate value - The feedrate value is the numerical value at which a tool will traverse a workpiece. It is usually expressed in either IPM (Inches Per Minute) or IPR (Inches Per Revolution).

Cutting speed - Cutting speed is the rotational speed of the cutting tool or workpiece. It can be stated as either RPM or SFM. (Surface Feet Per Minute)

Depth of Cut - Depth of cut is the distance the tool tip is engaged into the workpiece. It is incorporated into the X, Y, and Z values in a CNC program. Separate from the program, it can be expressed in inches or mm.

1.2.2.6 CNC Programming (G code)

G Code is used for CNC programming, the basic commands include the following 10 types:

- G00 - Point-to-Point Positioning
- G01 - Linear Interpolation
- G02 - CW Circular Interpolation
- G03 - CCW Circular Interpolation
- G40-G42 - Tool Diameter Compensation
- G70 - Inch Units
- G71 - Metric Units
- G80 -G89 - Canned or Fixed Cycles for Drilling
- G90 - Absolute Programming Mode
- G91 -Incremental Programming Mode

The figure below illustrates a typical programming sample:

```
%                    Program start Flag
: 1001               four digit  program number
N5 G90 G20           Use absolute units and inch programming
N10 M06 T2           Stop for tool change, use tool#2
N15 M03 S1200        Turn the Spindle on CW to 1200 rpm
N20 G00 X1 Y1
N25 Z0.125
N30  G01 Z-0.125 F5
N35 G01 X2 Y2
N40 G00 Z1
N45 X0 Y0
N50 M05
N55 M30
```

Figure 12 Sample of CNC Programming

1.2.2.7 Benefits of CNC Machining

The benefit of CNC machining include the following:

1. Increase production throughput;
2. Improve the quality and accuracy of manufactured parts;
3. Stabilize manufacturing costs;
4. Manufacture complex or otherwise impossible jobs - 2D and 3D contours.

1.3 Thesis Organization

This thesis is divided into 6 chapters and 4 appendixes:

Chapter one presents the background of this thesis, overviews related techniques.

Chapter two is a literature survey on the current works that are relative to e-Manufacturing. It also presents the objective and motivation of this thesis.

Chapter three gives a general description of the tools that will be used in the thesis.

Chapter four presents the 'Remote PC Access' approach, which can be used for utilization of remote resources, including the implementation on three manufacturing software: Catalyst, AutoCAD, and MasterCam.

Chapter five presents an Internet-based data exchange system, including its implementation and experimental results.

Chapter six draws the conclusions and gives recommendations for the future work.

Appendix A of this thesis provides the source codes for e-Manufacturing Data-Exchange System developed throughout the course of this study, including JSP, HTML Codes, XML Codes and JAVA Codes. Appendix B is the User Guide for using the created system. Appendix C is the output SML file, created by remotely using the Catalyst software. Appendix D is the output NC file, created by remotely using the MasterCam8.0 software.

CHAPTER 2 – LITERATURE REVIEW

Many attempts to achieve the goal of Internet-enabled Manufacturing have been reported in the literature. Some researchers presented methods to realize the remote control and monitoring of Rapid Prototyping via the Internet. Others attempted to utilize Internet for on-line activities, such as remote part measurement and visual simulation. Within this chapter, those attempts are discussed.

2.1 Attempts of Internet-enabled Manufacturing

The concept of distributed system has long been used in the manufacturing and design industries; however, traditional distributed system are hardwired and limited to a particular location and resources and accessibility is limited to a particular locations only. Recently, there is considerable interest in the area of Internet-enabled distributed system, and they are summarized as below:

2.1.1 Tele-Manufacturing Facility [Bailey, 1995]

The Tele-Manufacturing Facility (TMF) [Bailey95] is probably the first system that provides users with direct access to a rapid prototyping facility over the Internet. This system uses client-server software to handle the communication between the user and the prototyping facility. The user uploads CAD models through a WWW browser to the server. Upon receiving a CAD model, the server automatically performs error checking for the model file. It then puts the fabrication job into a queue of jobs waiting to be processed by a Helisys 1015 Laminated Object Manufacturing (LOM) rapid prototyping system.

2.1.2 Distributed Rapid Prototyping Via the Internet [Tay, 1999]

Remote rapid prototyping refers to the situation where the product model, which is designed at one geographic location, is transferred across the Internet to an RP system at another location for fabrication. The purpose of remote RP systems is to share the RP device among a larger community of engineers to dilute the cost of investment, similar to the network-connected printers that are shared by multiple users.

I-Man system [Francis, 1999] is an example of an early form of prototyping service on the Internet. To utilize the manufacturing service of this system, users need to convert their CAD models into a standard format and mail them over the Internet to the manufacturing facility. Human intervention is then required at the manufacturing site to pick jobs, setup the manufacturing system and perform job planning. This system provides an automated job queuing system, and it is up to the machine operators to decide the job processing order.

2.1.3 Internet-based Remote Control and Monitoring Rapid Prototyping System [Luo, Tzou, and Chang, 2001]

Ren C. Luo, Jyh Hwa Tzou, Yi C. Chang created a Internet-based Remote Control and Monitoring Rapid Prototyping System in 2001, which can perform provide a capability of on-line inspection. The algorthm they used include Display Algorithm of LCD Photomark, Pattern Matching Algorithm. The experimental tools include: Cad Tool, Photo-Mask, Curing Light source, SLP Rapid Prototyping Machine, Nepscape.

2.1.4 Web-based e-manufacturing system [Jin, Oraifige, Lister and Hall, 2001]

Java Applications, VRML Browser, External Authoring Interface (EAI), cad tools were used to create a Virtual System, It allows engineers and designers to visualize, explore, manipulate and interact with manufacturing applications in a Net-VEs.

2.2 The work of Francis E.H. Tay and his team

As the pioneer attempt to remote-control manufacturing, the approach was originally described as 'Internet Manufacturing' method [Tay, 1999]. Among the previous studies in the direction of Internet-enabled Manufacturing, Francis E.H. Tay and his team (1999) had a focus on remote control and monitoring of Manufacturing

Similar to Francis' research, my research focuses on the utilization of remote resources and Data exchange via the Internet. For this reason, Francis' will be introduced more detailed in this section, comparison can then be made with my methodologies, in order to highlight the difference and the gap my work will fix.

Here below is the summarization of Francis's research back in 1999:

2.2.1 Software and Hardware

Francis E.H. Tay and his group used the following software/ Hardware to achieve a remote-control system:

(A) QuickSlice™ Software

QuickSlice is a pre-processing software developed by Catalyst company, which is used in Fused Deposition Modeling (FDM). QuickSlice reads the .STL file generated by capable CAD/CAM systems and converts them into output files, which the FDM machine can use. An internal manufacturing file format created by QuickSlice is '.SML' file.

In Francis E.H. Tay's work, QuickSlice software is required by the client to convert STL files to SML files, which are recognized by the Rapid Prototyping machine. This software will reside on the sever side and be recalled remotely by the client through the use of the VNC application.

(B) Virtual Network Computing Software (VNC)

The technology underlying the VNC system is a simple protocol for remote access to graphical user interfaces. It works at the framebuffer level and therefore applies to all operating systems, windowing systems, and applications— indeed to any device with some form of communications link. The protocol will operate over any reliable transport such as TCP/IP.

The endpoint with which the user interacts (that is, the display and/or input devices) is called the *VNC client* or *viewer*. The endpoint where changes to the framebuffer originate (that is, the windowing system and applications) is known as the *VNC server*.

The following figure illustrates the architecture of Virtual Network Computing Software (VNC):

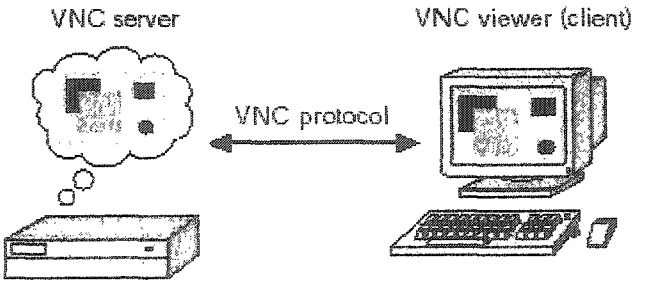

Figure 13 Architecture of VNC

This software component is invisible to the user of the Internet Manufacturing application. It resides on the server. In Francis E.H. Tay's work, the VNC software

functions as a server program, which allows the Internet Manufacturing users to access QuickSlice remotely.

(C) Java Development KIT (JDK)

A Java application was built with Java Development KIT (JDK). The Java application is the controlling program that allows the clients to use the other components smoothly and seamlessly.

(D) Fused Deposition Modeling machine (FDM)

Francis E.H. Tay had the approach implemented on a FDM machine. It is a physical machine that will do rapid prototyping.

The FDM technology is based on the layer-by-layer build-up process. After CAD model is converted to the STL file format, it is exported to the FDM HP workstation for preprocessing. This involves the step of slicing the model into thin cross-sections at the desired resolution anywhere from 0.051 to 0.762 mm using the Catalyst proprietary QUICKSLICE software. In the same process the supports are created if required and sliced. Thereafter, the sliced model and supports are converted into a Catalyst modeling language (SML) file that directs the paths of the nozzle called "roads" that deposit the extruded thermoplastic material to create each cross-section layer-by-layer.

The following figure illustrates a Fused Deposition Modeling machine:

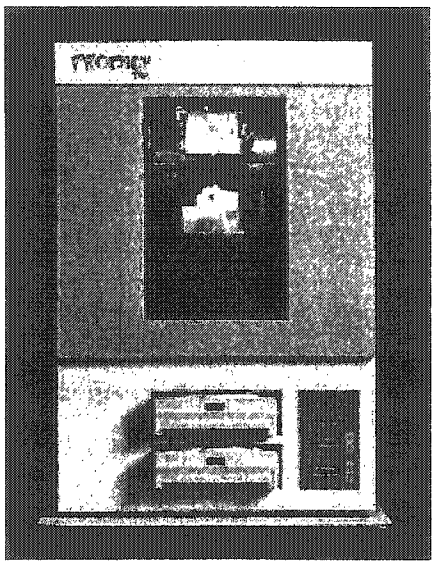

Figure 14 Picture of a Fused Deposition Modeling machine

2.2.2 System Architecture

The Java-based Sever is the core of the system, it 'talks' with the Remote Sever (Client Side) and the machine Sever. The following figure illustrates the system architecture of the Control:

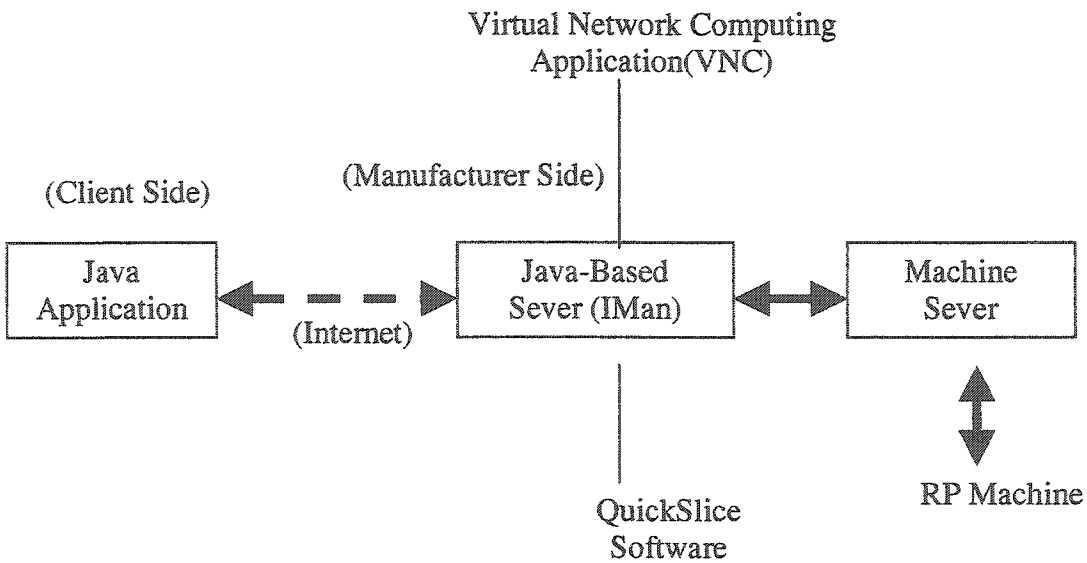

Figure 15 System Architecture of Francis' Research [Francis, 99]

2.2.3 Major Functionality

Using 'Software Interface' approach, a remote user could be do the following tasks via the Internet:

- Transfer a STL file;
- Use QuickSlice software remotely tot create SML file;
- Remote view the Status of the Machine;
- Activate the machine to process a existing SML file.

2.2.4 The limitations of the approach

In comparison with regular practice, the core stuff Francis E.H. Tay and his team added into the remote-control system is the Java-based Sever, which is designed to communicate to the remote sever through Internet and the local manufacturing machine. In Francis E.H. Tay's work, the Java Server was specifically built to 'talk' to a specific type of manufacturing software (e.g., QuickSlice). Therefore, there exists a couple of limitations with this approach:

- **Commonality Issue**- The sever is software-depended, in other words, because the sever is built to 'talk' to QuickSlice software, it can't be used to other types of Rapid-Prototyping, which use different manufacturing software;
- **Data-based Issue** - Because the sever was built only as a 'communicator', it would not have the following functionality: store data for further tracing / analysis / feedback / fault detection, prognostics, etc.
- **Firewall Issue**- The Connection of the two computers cannot get through the firewall.

2.3 Francis E.H. Tay 's Research vs. My Research

The brief comparison below has showed the major difference between Francis E.H. Tay's work and my approaches in Internet-Manufacturing:

Table 3 Research comparison with Francis (1999)

	Focus On	Methodology	Gap or What's New
'Internet-Manufacturing System ' [Francis, 99]	Remote control and monitoring of Rapid Prototyping Machining through Internet	A user platform was created with Java programming tool	Gap- 1. The platform is not common, because it is programmed to 'talk' to one specific software 'Quickslice' 2. The system is 2-tier only (User - Internet)
'PC Remote Access' Approach (Figure-18, 19)	Utilization of resource at remote site through Internet	Use of commercial software called 'PC Remote Access'	New- 1. It is more common, because it doesn't depend on software-related programming 2. Improved with software development
'Data-Based Internet Manufacturing' System (Figure-28, 29)	The exchange of manufacturing data through Internet	User platform created with JSP, Java, and Oracle Database	New- 1. It is 3-tier (User – Internet - Database). It can be accessed from anywhere globally, due to its JSP-based feature 2. Data-transaction function (Storage, Search, Delete)

2.4 Thesis Objective

Review of the previous work in Internet-enabled Manufacturing shows that many areas have been extensively researched in specific cases. However, it is far from the end of research, there are still some shortcomings which may limit the application of the approaches, for example, in Francis' research, there are several issues which need to be addressed (refer to section 2.2.4, limitation of Francis' research). An Internet-enabled system should be more common, more predictive, and more data-based. In this research, effort will be made to identify methods to have the Internet-enabled system more commonality and more data-based feature.

To this end, several objectives have been defined:

- To develop techniques and guidelines for creating an Internet-based platform;
- To connect a database to the system, which may add the feature of 'Data storage, search and delete';
- To identify a more common way to realize the 'Remote Access' via the Internet (Refer to section 2.2.4, limitation of Francis's research)

CHAPTER 3 – EXPERIMENTAL SOFTWARE TOOLS

In this section, we provide a brief introduction to the major software we will use for the proposed approaches, 'Remote PC Access' and 'Data-based Internet Manufacturing' system.

3.1 Java Programming Language

3.1.1 What is Java

Java is a programming language first released by Sun Microsystems in 1995, which allows World Wide Web pages to contain code that is executed on the browser. Java is an object-oriented programming language with a built-in application programming interface (API) that can handle graphics and user interfaces and that can be used to create applications or applets. Because of its rich set of API's, similar to Macintosh and Windows, and its platform independence, Java can also be thought of as a platform in itself.

More than two million programmers have learned the language, and are using it in places such as NASA, IBM, Kaiser Permanente, ESPN, and New York's Museum of Modern Art. It is a standard academic curriculum at many computer science departments around the world. More than 1,700 books have been written about it, according to the most recent *JavaWorld Magazine* count.

First used to create simple programs on World Wide Web pages, Java can be found today in each of the following places and many more: Web servers, Relational databases, Mainframe computers, Telephones, Orbiting telescopes, Personal digital assistants.

3.1.2 How Java Works

Java is an object-oriented, platform-neutral, secure language.

Object-oriented programming

It is a software development methodology in which a program is conceptualized as a group of objects that work together. Objects are created using templates call classes, and they contain data and statements required to use the data;

Platform-neutral

It is the ability of a program to run without modification in different computing environments. With the compiler, first user translates a program into an intermediate language called *Java bytecodes* —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java bytecode instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed.

The following figure illustrates how this works.

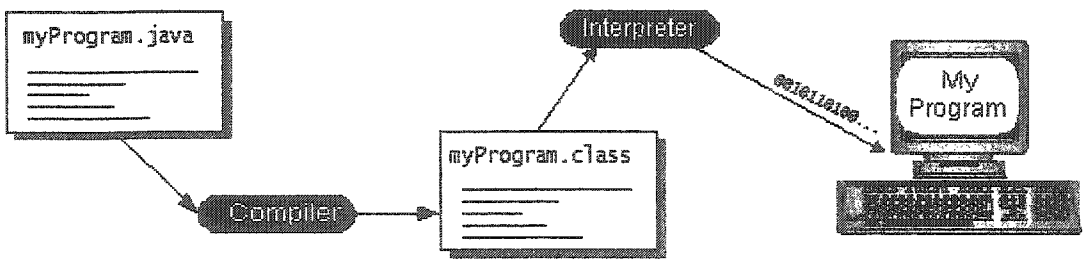

Figure 16 illustration of how Java works

3.1.3 Java Software Development Tool

There are several popular integrated development environments for Java, including:

- Java 2 Software Development Kit

- Borland Jbuilder
- WebGain Visual Café
- Sun ONE Studio
- IBM VisualAge for Java

In this thesis, Java 2 Software Development Kit is utilized for the finished work.

3.1.4 Important concepts of Java Object and Class

- **Object**

An *object* is a self-contained element of a computer program that represents a related group of features and is designed to accomplish specific tasks.

- **Class**

A *class* is a template used to create an object. Every object created from the same class will have similar, if not identical, features. The class library handles numerous tasks, such as mathematical functions, text handling, graphics, sound, user interaction, and networking.

- **Attribute**

A Java class consists of two distinct types of information: *Attribute* and *Behavior*. *Attributes* are the data that differentiate one object from another. They can be used to determine the appearance, state, and other qualities that belong to that class.

- **Behavior**

Behavior refers to the things that a class of objects can do to themselves and other objects.

- **Methods**

Methods are groups related statements in a class of objects that handle a task. They are used to accomplish specific tasks on their own objects and other objects, and they are used in the way that functions and subroutines are used in other programming languages.

- **Inheritance**

Inheritance is a mechanism that enables one class to inherit all the behavior and attributes of another class.

- **Methods**

Methods are groups related statements in a class of objects that handle a task. They are used to accomplish specific tasks on their own objects and other objects, and they are used in the way that functions and subroutines are used in other programming languages.

3.2 Oracle Database Management System (RDBMS)

3.2.1 Brief Introduction of Oracle Database

The Oracle Relational Database Management System (***RDBMS***) is an industry leading database system designed for mission critical data storage and retrieval. The RDBMS is responsible for accurately storing data and efficiently retrieving that data in response to user queries. The Oracle Corporation also supplies interface tools to access data stored in an Oracle database. Two of these tools are known as SQL*Plus, a command line interface, and *Developer/2000* (now called simply *Developer*), a collection of forms, reports and graphics interfaces.

3.2.2 Brief Introduction to Application Development Tools

1. SQL*Plus - A command line tool used to manipulate tables and other database objects in an Oracle database.
2. Developer/2000 and Developer A suite of application development tools including Forms, Reports and Graphics.
   - Oracle Forms - A screen based tool used to develop data entry forms and menus that access tables in an Oracle database.
   - Oracle Reports - A screen based tool used to develop reports that access tables in an Oracle database.
   - Oracle Graphics - A graphical tool used to develop charts and reports that access tables in an Oracle database.
3. Oracle JDeveloper - A general purpose Java Integrated Development Environment that has been pre-loaded with classes and methods used to connect to and manipulate schemas in Oracle databases. A collection of code development wizards allow the developer to quickly create data entry forms as Java applications or applets as well as reports using Java Server Pages (JSP).

The Developer tool set was initially aimed at developing traditional two tier client/server applications where the client side holds the forms and reports user interfaces as well as the majority of the business logic. Business logic is implemented using Oracle's PL/SQL language. These tools have steadily been revised to fit a three-tier architecture where the client only processes user interface elements while a "middle tier" takes care of business logic processing.

3.2.3 Typical Development Environment

Developing applications using an Oracle database requires access to a copy of the Oracle RDBMS (or a central Oracle RDBMS server), and one or more of the development tools. Third party development tools such as PowerBuilder, Visual Basic or Java can also be used for applications development.

3.2.4 SQL Language

Structured Query Language (SQL) is the language used to manipulate relational databases. SQL is tied very closely with the relational model.

In the relational model, data is stored in structures called relations or *tables*. Each table has one or more attributes or *columns* that describe the table. In relational databases, the table is the fundamental building block of a database application.

SQL statements are issued for the purpose of:

- Data definition - Defining tables and structures in the database (DB).
- Data manipulation - Inserting new data, Updating existing data, Deleting existing data, and Querying the Database (Retrieving existing data from the database).

3.3 I-DEAS 8.0

I-DEAS is an integrated package of Mechanical Engineering software tools.

I-DEAS is composed of a number of software modules called applications: Design /Drafting / Simulation/ Test/ Manufacturing/ Management.

In the work, the I-DEAS is used to create the solid part (modeling).

3.4 PC Remote Access 1.0

3.4.1 What is Remote PC Access?

Remote PC Access is a software program that lets you access your PC from another PC via any internet or LAN connection. It also allows system administrators to take control of any computer on their system, and it lets computer support personal access files, take control of remote computer's keyboard and mouse to solve problems (with permission, of course, of the remote computer owner). Remote PC Access software works over any networks which supports standard TCP/IP protocol (this includes the Internet, corporate networks, home networks, and direct PC to PC phone connections).

3.4.2 What is its functionality?

The following figure illustrates a typical method of remote access:

The functionality includes:

- **Access** — Remotely access a computer from any PC with a Web browser . There is no additional software to install - A self-launching plug-in will allow you to see your host computer.

- **Work** — Begin working on your host computer as if you were sitting in front of it. You can immediately access your email, applications, documents and network resources.

The following figure illustrates a typical method of remote access:

:

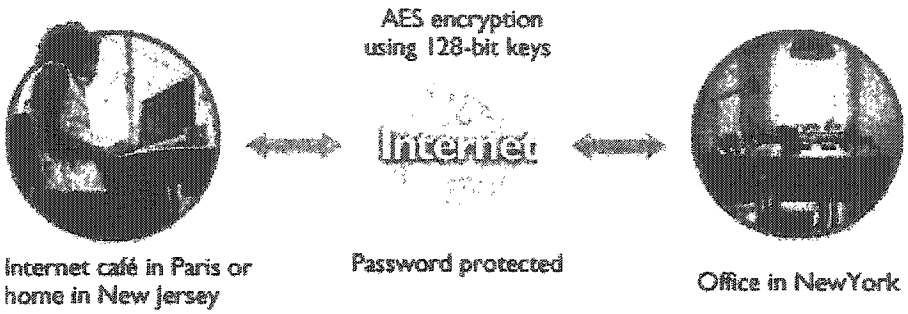

Figure 17 Illustration of a typical method of remote access

With this software, a user can grant access to his PC to any number of users. For each user he can create a separate account with a separate username and password and specify any of the following access rights:

- View screen
- Control keyboard and mouse
- Access file system, which can be further restricted to:
    1. Specific folder
    2. Upload
    3. Download
    4. Delete

A user can specify access rights for each user when he creates user accounts. He can change the access rights at any time by editing a user account on the Users and Passwords page of the Remote PC Server window. Clients connected to his PC are listed in the Active Connections tab of the Server's window. The user can disconnect an active connection by selecting it and pressing the Disconnect button.

3.4.3 About firewall...

The current major problem of this software is that it cannot help a PC to get access to another PC behind Router or firewall. Therefore it will limit its application in Industry. However, here is a notice on product platform: "Coming soon: sound transfer, transparent operation with NAT routers and firewalls, support for dynamic IP addresses". We can then expect a transparent software version in the near future.

CHAPTER 4 – IMPLEMENTATION OF 'REMOTE PC ACCESS' METHOD

4.1 Introduction

Francis E.H. Tay's and his team (1999) had already demonstrated their user platform, which was built up mostly with Java language. The original consideration of the methodology 'PC Remote Access' was also to achieve the remote control of monitoring of a manufacturing process, by using available resource. However, the firewall-related weakness of the 'PC remote Access' software has denied the feasibility (See the introduction of this software in chapter 3). As a tradeoff, in this thesis, this methodology was tested for utilization of remote resources.

This method is different from the previous attempts, in that it uses pure commercial software to build up the connection between local machine and remote users, instead of by the programmed platform.

4.2 Hardware/ Software and System Architecture

This approach uses the following tools:

(1) Hardware-

Two PC computers with Internet access, their Windows version must be one of following: 95/98/Me/NT/XP/2000.

As introduced in chapter 3, the firewall blocks the connection with 'PC Remote Access' software, so the controlled computer in thesis demonstration is the one without firewall protection. In the experiment we did, the controlled side is a computer at off-campus location, and the controlling side is computer located on campus in Essex Hall building of University of Windsor.

(2) Software-

- IDEAS CAD/CAM/CAE
- PC Remote Access 1.0
- Catalyst Prototyping Software
- AutoCAD 2000
- MasterCam8.0

The following figure shows the architecture of remote utilization of Catalyst Prototyping Software via the Internet. At the remote side, a third facility (printer or Manufacturing tool) could be connected for outputs.

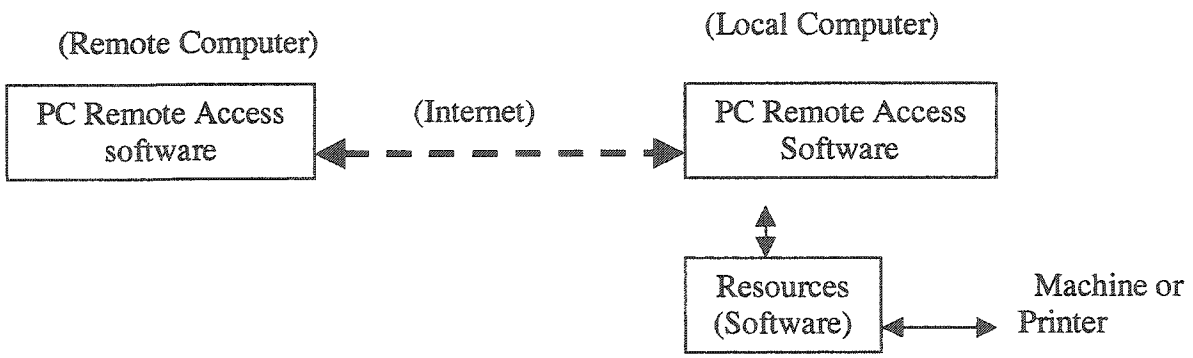

Figure 18 Architecture of remote utilization of Resource

And the following diagram shows typical input and output of remote utilization of Rapid Prototyping software via the Internet.

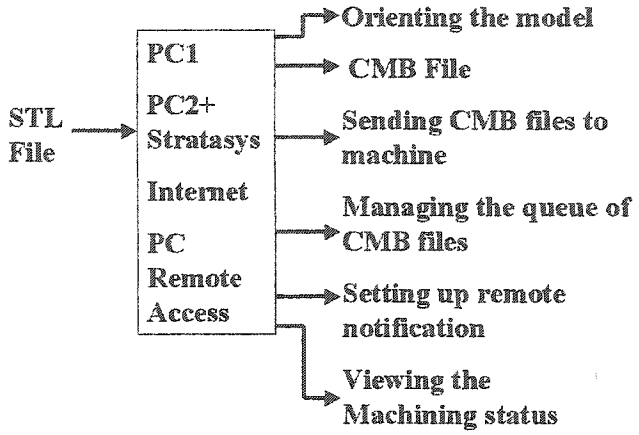

Figure 19 Input / Output of remote utilization of Rapid Prototyping software

4.3 Experiment Setups/ Steps

We assume that local user will use this method to utilize the remote resource (Software) via the Internet. The functionality include includes file transfer and using a software remotely.

In order to have anticipated result, here is a summarization of experiment setups:

1. Have 2 computers connected to Internet, one at local side, one at remote side; The computer at remote side must be a computer without firewall protection;
2. Remote computer- Have the required resources (software) installed, for example: Catalyst, AutoCAD2000, or MasterCam8.0;
3. Local computer- Download and install the 'PC remote Access 1.0' software from the web site www.download.com.;
4. Remote computer- Download and install the 'PC remote Access 1.0' software from the web site www.download.com;
5. Remote computer- Turn on the 'Remote PC Sever' by clicking START-PROGRAMS-ACCESS REMOTE PC1.0-REMOTE PC SEVER;

6. Remote computer- From the control platform, notice and mark down the IP address / port # of the computer, for example: IP=24.57.36.175/Port=34012; This information will be used by user at local computer;
7. Remote computer- From the control platform, add an 'user and password' by clicking the button 'user and password', and assign an user name/ password, for example: 'marvin/marvin'. This information will be used by user at local computer;
8. Local computer- Turn on the 'Remote PC Client' by clicking START-PROGRAMS-ACCESS REMOTE PC1.0-REMOTE PC CLIENT.

After the steps of through 1-8, the setup is done. In order to have anticipated result, here is a summarization of experiment steps:

1. Local computer-Connect the remote computer, by clicking 'Connect-Connect to' on platform; Type in the IP address and port # (see setup step 6);
2. Local computer- After about 10 seconds, a prompt will request User name and password. Type in the User name and password (see setup step 7);
3. Remote computer- After about 10 seconds, a prompt will request approval of access by local user. Type YES for approval.

After the steps of 1 to 3, connection is done, and screen of remote computer is shown in the screen of local computer.

4.4 Implementation one– Remote Utilization of Catalyst Software

In March 2003, tests were carried out at an off-campus location. The 'PC Remote Access' software was installed on two PC computers beforehand, and Catalyst software was installed in one of the computers, the 'local machine'.

The remote control and monitoring of a Rapid Prototyping was realized by using the 'PC remote Access' software simultaneously on the two computers. Here are the steps of the implementation and some demonstrations:

Step 1- Installation of 'PC Remote Access' on two PC computers

There are some requirements for the used computers:

- The operating system of the two computers are Windows system;
- They can be connected to the Internet;
- The computer working as 'local computer' must have Catalyst software installed.

Step 2- Connect both computers to the Internet.

Step 3- Run 'PC Remote Access' software on Remote computer

Remote operator has to input the password for this action.

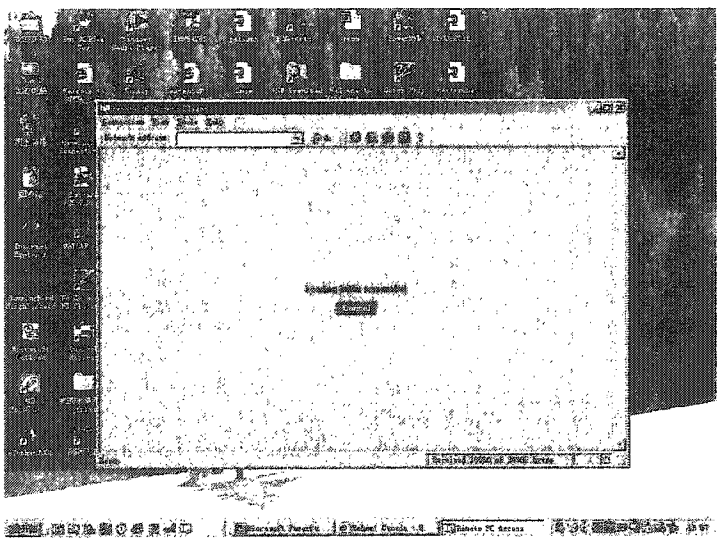

Figure 20 Remote computer is starting 'PC Remote Access'

Step 4- Remote user remotely runs the Catalyst software in local computer

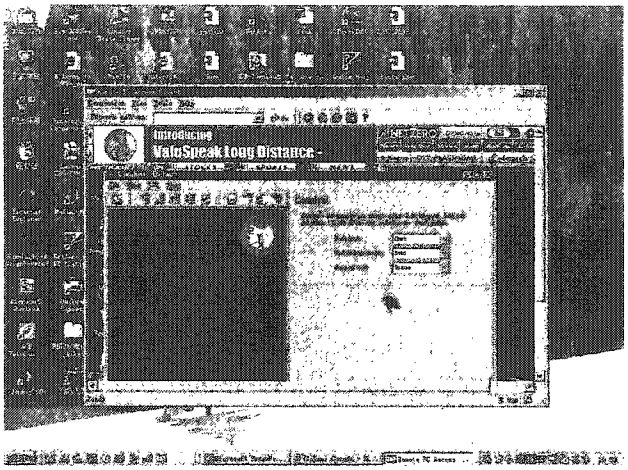

Figure 21 Remote user running the Catalyst remotely

Step 5- Remote user open a STL file remotely

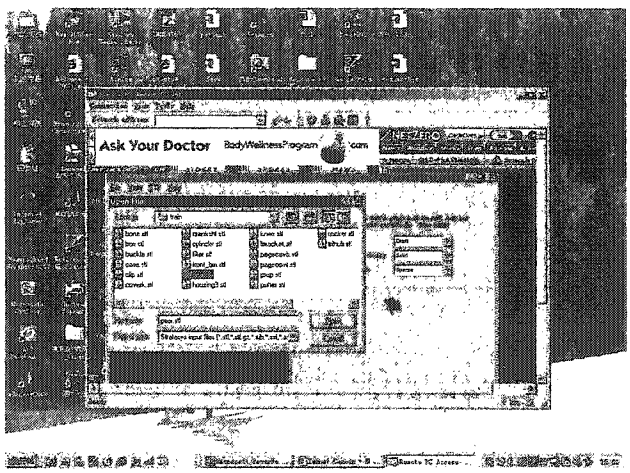

Figure 22 Remote user open a STL file remotely

Step 6- Remote user transfer STL file to SML file remotely

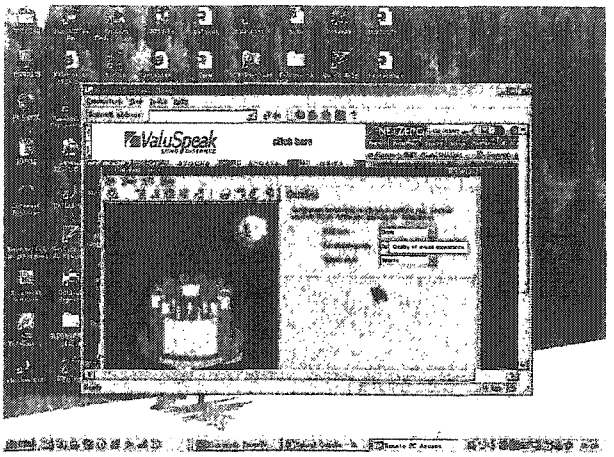

Figure 23 Remote user transfer STL file to SML file remotely

The created SML file can be viewed at Appendix C.

Step 7- Remote user starts the manufacturing remotely and see the simulation

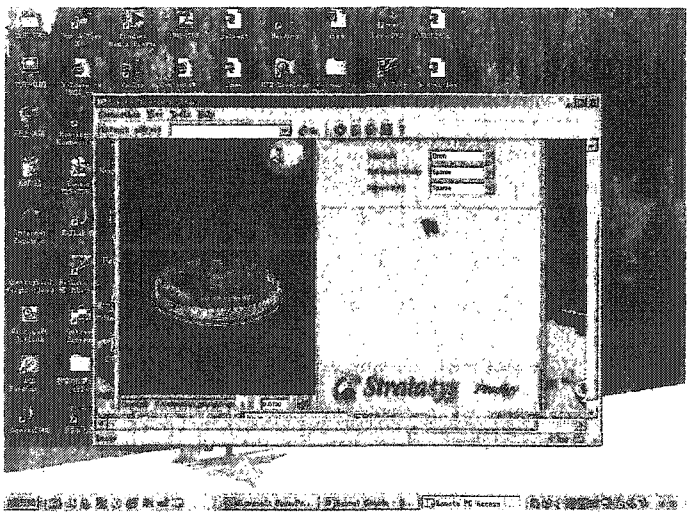

Figure 24 Remote user transfer STL file to SML file remotely

4.5 Implementation Two– Remote Utilization of AutoCAD 2000

Tests were also successfully done to remotely use the AutoCAD 2000, a 'Gear' drawing was made remotely and printed out. The following figures shows the screenshot and finished drawing.

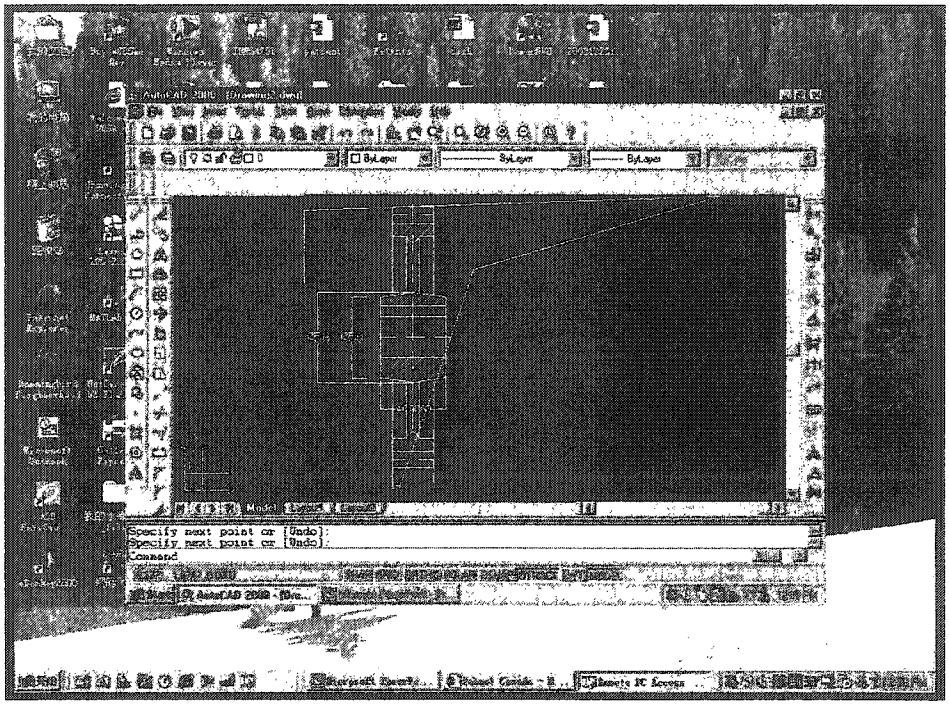

Figure 25 Remotely using AutoCAD2000 via The Internet

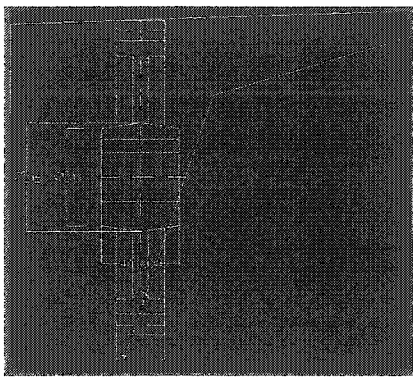

Figure 26 'Gear' drawing remotely created & printed out via the Internet

4.6 Implementation Three– Remote Utilization of MasterCam 8.0

Tests were also successfully done to use the MasterCam via the Internet, the following operations were carried out remotely:

1. Loading an Model file from remote computer side;
2. Run the Software remotely;
3. Output the NC manufacturing file (See appendix D).

The screenshot of implementation is shown in the following figure:

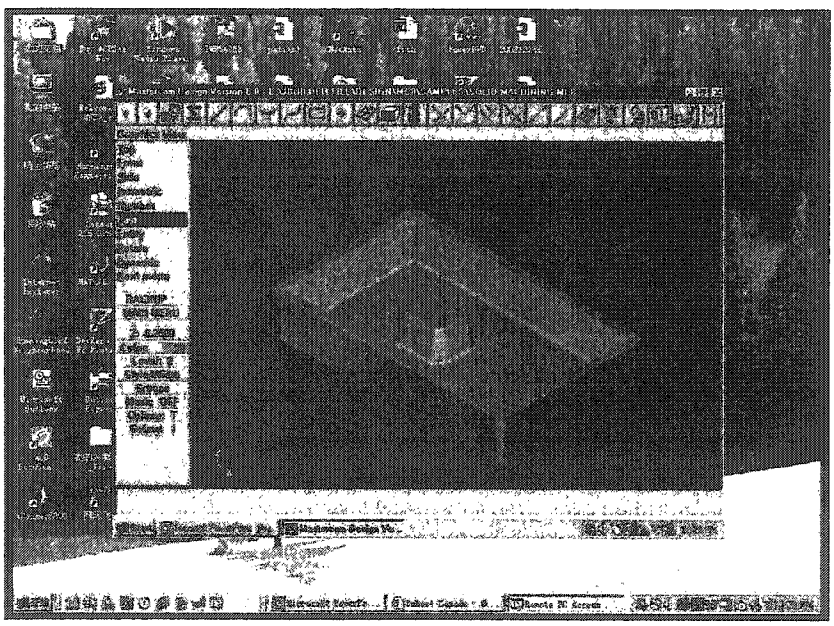

Figure 27 Screenshot of Remote Utilization of MasterCam8.0

4.7 Results and Discussion

4.7.1 The results

Tests were successfully done to use the software through the Internet.

1. The operation was proved to be simple;
2. No errors were returned, and no special situations were encountered;
3. The results of remotely using software were achieved, mainly including:
   - Remotely using Catalyst ->SML files was created; The created SML file can be viewed at appendix C
   - Remotely using AutoCAD2000->Drawings were made and printed out; The created drawing is shown as Figure 26
   - Remotely using MasterCam->Toolpath / NC files were created; The created NC file can be viewed at Appendix D
4. Tests also show that 'remotely using the software' is a little slower than using it directly.

4.7.2 The limitations

However, it needs to be pointed out that some limitations exist with this approach, due to the software itself:

1. When the software was used remotely via the Internet, the speed of running the software was not as fast as running it directly, probably because of the additional data transition through the Internet;
2. It is platform-depended, e.g., the computer operating system must be Windows;
3. Local computer's firewall, if any, will block the remote access with the existing software, because of the Internet security; this could be a major problem for further application, because most of the major industrial plants, like GM, Ford, have the firewall at computer systems;

4. Because in this control, the local computer system would be 100% exposed to the remote user, it may in return limit the its application in the global cooperation among Clients/Manufacturers/ Suppliers/ Partners.

4.7.3 Recommendation for further experiments of this method

As introduced in chapter three, in the near future, the software 'PC Remote Access' will fix its limitation in dealing with firewall. After the solution of this bottleneck, more successful experiment could be done to achieve a 'Internet-enabled Manufacturing system'. For example, experiments can then be done in the University of Windsor, to have the Prototyping machining in lab B27 (Essex Hall) controlled and monitored remotely, by any computer with Internet Access. Similar experiments could also be considered to be in the area of Robot-related Manufacturing.

4.7.4 Application of this methodology in other areas

From the test, we can see that this method has the potential to enable the remote control and monitoring of a real processing, if the firewall bottleneck can be fixed.

Besides of its potential in Manufacturing, a user can also get benefit in some other ways, like:

1. Individual Use - control computers in home network, access a home computer from office or work on the office computer from home, and access the office or home computer while traveling. Invite someone to his PC to fix it remotely.
2. Business Use - The main office can access branch office desktops, or the branch office can access main office server. The employees can access email, files, programs, and network resources from home or on the road. System administrators can remotely access and manage multiple PCs from anywhere. Sales force can access applications and documents and continue working while on the road.

CHAPTER 5 – INTERNET-BASED DATA EXCHANGE SYSTEM

5.1 Introduction

In this chapter, a novel methodology called 'Data-based Internet Manufacturing' is proposed, and implementation and results are also demonstrated accordingly. The 'Data-based Internet Manufacturing' was developed to remedy the shortcomings existing in the previous attempts. Such shortcomings include the lack of storage of manufacturing data, and connections of the input data.

The core task of this methodology is to build up a Java-based server and link to a database. The main features of this server will include the following:

- Because it is built with Java, the sever is not software-depended or hardware-depended, it is 'neutral'. In another word, this server can be used on any operating system, and 'talk' to any manufacturing software;
- Because the server is linked to an Oracle database system, it is capable of storing manufacturing data, and more important, the global users (suppliers, clients, and partners, etc.) can get access to necessary data of the database, based on their assigned priority level.
- When sever can also provide a function of 'degradation prediction' and 'prognostics', if it is incorporated with some algorithm.

5.2 Overview of 'Data-Based Internet Manufacturing' System

'Data-based Internet Manufacturing' integrates Oracle Database into a Java-based platform. It offers the user the advantage of 'storing data' and better data connection over the previous Java-Platform [Tay, 1999].

The developed system utilized the JSP (Java Sever Page) language to create the necessary web pages, and an Oracle Database was linked to the platform for data storage

purpose. Therefore, the input of the system is the raw data upload or input from the system users (vendors, manufacturers, customers, etc.). The output of the system could be the search results, or the downloaded data from the system. In the meantime, password access is also database related, so only the authorized users can get access to the required data from the system.

5.3 Structure of 'Data-Based Internet Manufacturing'

The concept of the 'Data-based Internet Manufacturing System' is to provide the user a platform to search information, or receive required precaution when it is necessary. Accordingly the use of the designed system include the following aspects:

- Password Access;
- Upload/ Download data, such as Machining files (for example, NC format), Process Monitoring Video;
- Input data, such as material property, shipping information, etc.;
- Search/ Delete Data.

The following figure demonstrates the input/out relation of the system:

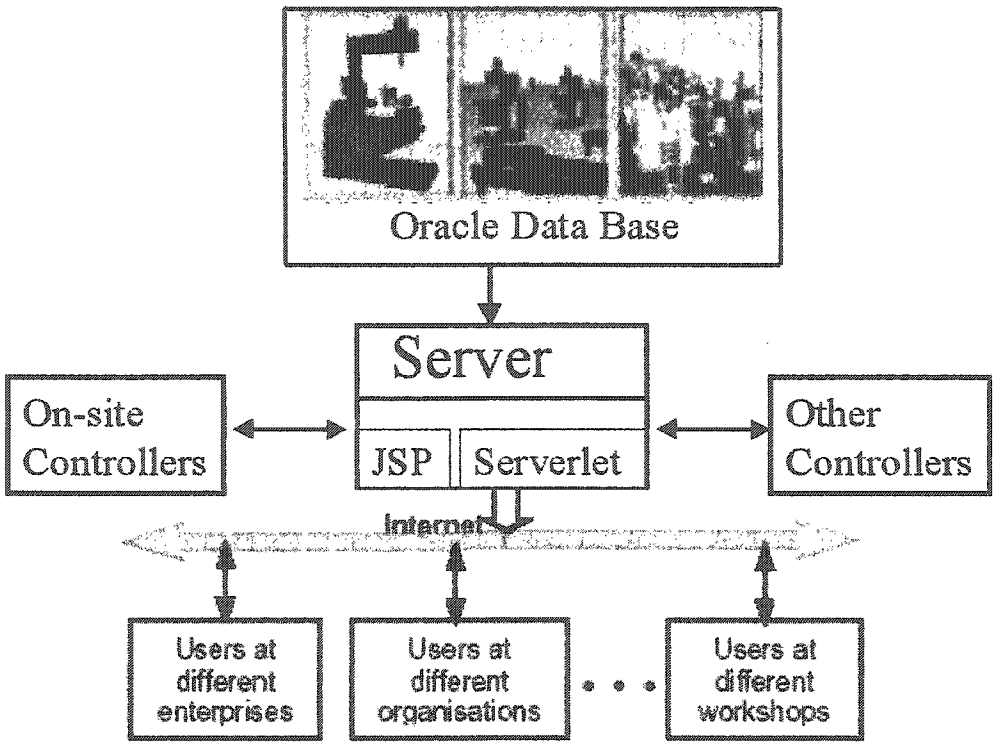

Figure 28 Architecture of Implemented Approach

5.4 Experiment Setups and Steps

The Data-based Internet Manufacturing platform is created for global user to exchange manufacturing data. The functionalities include data input/ output/Search/ Delete/ Download/Upload.

In order to have anticipated result, here is a summarization of experiment setups:

1. Have a local computer ready and connected to the Internet;
2. Download and install the Tomcat Sever on local computer;
3. Have all the source codes in the Tomcat/Bin folder;
4. Have the Oracle database resource available, and have the user name and pass word information in the ACCESS Java file;

5. On the local computer, at the doc environment, go to tomcat folder, by typing: XX/tomcat4/bin;
6. Startup the tomcat severe at the required port, by typing for example 'startup 47026';
7. Now the platform is turned on; and the created web page can be used at the website: http://davinci.newcs.uwindsor.ca:47026/myproject/index.html.

5.5 Implementation Demonstration

The main factor for the validation is the fact that the created system could achieve the functionality that the previous system could not finish, including Password Access, Data-exchange within database. In order to demonstrate a sample system, Java-Based Platform was built up and Database (Oracle) was linked. The codes are attached in the appendix for further review, including html codes, xml codes, jsp codes and java codes.

5.5.1 Basic Operation Demonstration

The user guide for using the system is attached at Appendix B for further review.

The following figure shows the basic steps of how to use the sample system:

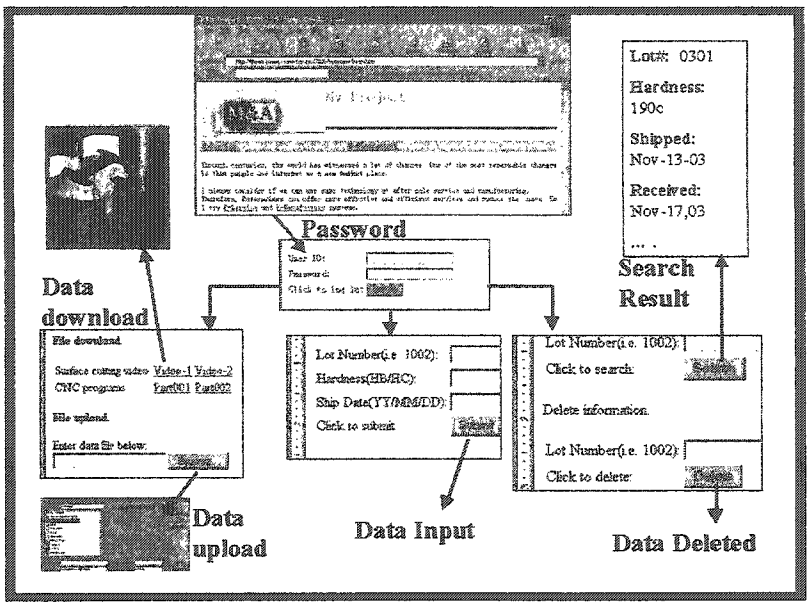

Figure 29 Basic operation of the created system

5.5.2 System Features

The following features could be successfully demonstrated with the created system:

(A) Password Access

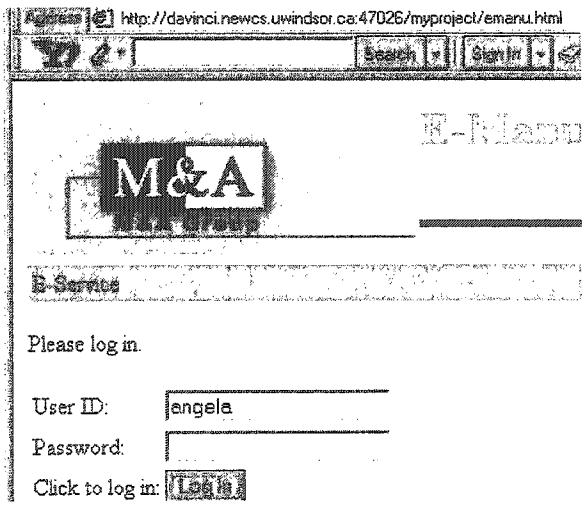

Figure 30 Password protection feature

(B) Data Upload/Download

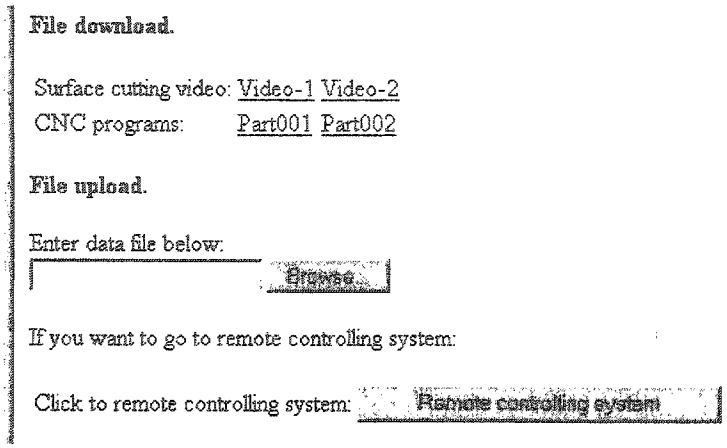

Figure 31 Data upload/download feature

(C) Data Input to Database

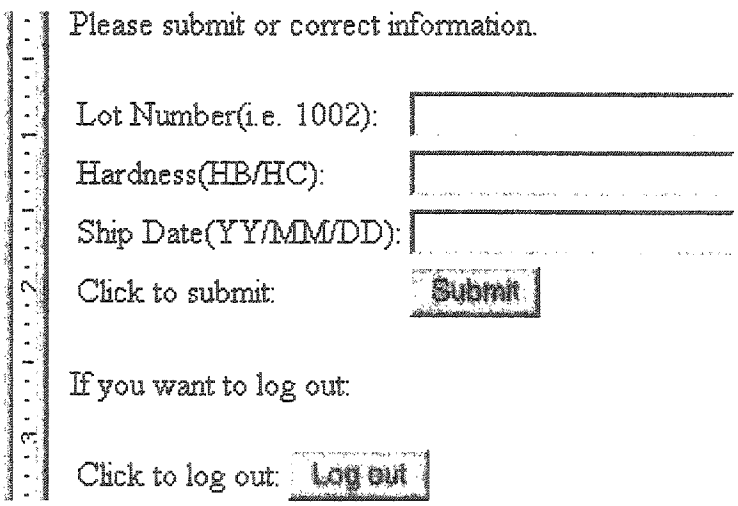

Figure 32 Data input into database

(D) Data searched or deleted within database

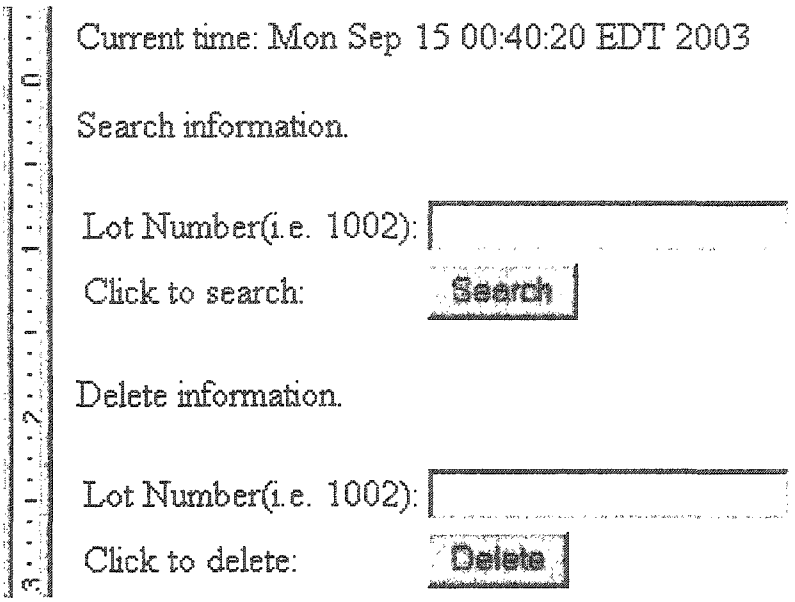

Figure 33 Data searched or deleted within database

5.6 Results and Discussion

5.6.1 The results

Platform was created and the tests were successfully done to use the platform to achieve its functionality (see section 5.5.1);

1. The operation was proved to be simple;
2. No errors were returned, and no special situations were encountered.

5.6.2 Recommendation for application of this method

The original concept of this methodology is to identify a novel solution to build up the platform, which can be used for global cooperation. Not too much fancy function were added into this prototype platform. In the future, platform designer can also consider add more specific application to enhance its functionality in manufacturing. For

example, an assembly supplier can change the web pages by adding the following data items:

- Product #, Purchase Order #
- Assembly Drawing Upload/Download
- Product Assembly Instruction Video
- Plant layout Update
- Etc.

CHAPTER 6 – CONCLUSION

6.1 Contributions

This thesis has made the following contributions:

1. Literature review in Internet-enabled Manufacturing was done to identify the current works that have already been done, and the methods that have been developed for the Internet Remote control and Monitoring of manufacturing.
2. A novel method for manufacturing remote control, 'Remote PC Access', was proposed and tested, and the limitation was also summarized.
3. 'Data-based Internet Manufacturing' was also proposed, which has addressed the shortcoming of data exchange in the previous attempts. This method utilized the JSP techniques and Oracle Database resource. Demonstration implementation was also done accordingly.

6.2 Conclusions

The Internet is a powerful tool that can be exploited for engineering use. The remote control and utilization of manufacturing and prototyping facilities have been implemented to enable the sharing of resources [Tay, 1999]. The purpose of this thesis is to identify the methodologies for Internet-Manufacturing.

A 'Remote PC Access' method was proposed, and presently it would work for the scattered manufacturing situation, without firewall protection. In the implementation, by using the 'Remote PC Access' software, experiment has been successfully done in file transfer and utilization of resource at remote site through Internet. The second method, 'Data-based Internet Manufacturing System, has improved the Java-based Platform by adding an Oracle database. The approach is targeted at the wide range of users in product life cycle, including manufacturers, venders, customers and manufacturing partners. In particular, platform independence is a critical factor to ensure that end users using different computer systems can interact with the data exchange system effectively. The

architecture for the Data-based Internet Manufacturing system is also highly scalable for further research and development. With this system, it becomes possible to collaborate in real time between two geographically distributed resources (partners), by using the cheapest data carrier, Internet, as the means of communications. In the global manufacturing and design arena, industrial competition through successful product development based on effective data exchange can be further enhanced by the use of Data-based System via the Internet to considerably reduce the product development lead time.

6.3 Recommendations for Future Work

The purpose of Internet-enabled Manufacturing is to provide a system where a wide range of users can share the resources in real time to handle manufacturing process faster and more effectively. To this mean, a few systems are already created and tested, among which include the Java-Bases Platform [Tay, 1999] and Data-based Internet Manufacturing System.

A number of issues, which provide future research related to this work, have been identified:

1. The firewall is the bottleneck for the application of 'PC Remote Access' method, efforts should be made to address the problem;
2. Intelligence algorithm can be added to the system and enable the system to predict an unusual quality situation, by sending PC-to-Phone message to users. For example, cutting tool duration algorithm could be treated with the tool usage data in the data base;
3. Automatic data entry to the system can be a great improvement to the existing Internet Manufacturing system.

REFERENCES

Bailey, M. J., (1995), *'Tele-Manufacturing: rapid prototyping on the Internet'*, Computer Graphics and Applications, IEEE, Volume: 15 Issue: 6 , Nov. 1995, Page(s): 20 –26

Bailey, M. J., (1995), '*Tele-manufacturing*', Computer Graphics and Applications, IEEE On page(s): 20-26 Volume: 15, Nov 1995

Bailey, M. J., (1996), '*Tele-Manufacturing:Rapid Prototyping on the Internet with Automatic Consistency-Checking*', San Diego Supercomputing Center (SDSC) Tele-Manufacturing Facility (TMF), http://www.sdsc.edu/tmf/Whitepaper/whitepaper.html

Bailey, M. J., and Clark, D., (2001), '*Using Chroma Depth to obtain Inexpensive Single-image Stereovision for Scientific Visualization*', Graphics Tools. Manufacturing Systems 2001, Volume: 12 Number: 6, pp: 409-415

Brady, K., and Tarn, T.J., (1998), '*Internet-based Remote Tele-operation*', Proceedings of 1998 IEEE international Conference on Robotics and Automation, Leuven, Belgium, May 1998

CNC Educational Website - http://users.bergen.org/jdefalco/CNC/

CNC Inspection Website- http://www.asktechnologies.com/cnc_inspection_station.htm

Dr. Emilio Olmos and Dr. Carlos López's Website- http://www.chez.com/webadvertising/About%20the%20site.htm

Gorbach, Greg, (2000), '*Collaborative e-manufacturing*', Manufacturing Systems; Wheaton; Sep 2000; Volume:18, Issue: 9, pp:116-117

Gorbach, Greg, (2000), '*E-Manufacturing Solutions*', Manufacturing Systems; Wheaton; Dec 2000, Volume: 18, Issue: 12, pp: 52-56

Gorbach, Greg, (2000), '*Seven Fundamentals of e-Manufacturing*', Manufacturing Systems; Wheaton; Nov 2000; Volume: 18, Issue: 11, pp: 120-121

Gordon, L.A., (2000), '*The Industrial Internet: a Total Digital Enterprise Mindset*'. USDLA United States Distance Learning Association Journal, On-Line Issue http://usdla.org/ED magazine/illuminactive.JUN00 Issue.story01.htm

Gordon, L.A., (2002), '*The e-Skip General Effect. The Emergence of Acybercentric Management Model and the F2B Market Segment for Industry*', International Production Economics, pp11–29

Grimaldi, D., Nigro, L., and Pupo, F., (1998), '*Java-Based Distributed Measurement Systems*', IEEE Transactions on Instrumentation and Measurement, Vol. 47, No.1, pp.100-103.

Kevin, Roach, (2002), '*Living Lean- E-manufacturing Promises to be the Next Productivity Realization*', http://www.google.ca/search?q=cache:S9ddo8M3YDAJ:www.plantservices.com/Web_First/ps.nsf/ArticleID/DPIC569KCN/+Kevin+Roach,+eManufacturing&hl=en&lr=lang_en&ie=UTF-8

Koc, Mummer, Ni, Jun, and Lee, Jay, (2002), '*Introduction of e-Manufacturing*', The 5th International Conference on Frontiers of Design and Manufacturing (ICFDM '2002)

Lee, Jay, (2001), '*Tether-free Technologies for e-Manufacturing*', http://www.imscenter.net%20<http://www.imscenter.net

Lee, Jay, and Koc, M., (2002), '*E-Manufacturing and e-Maintenance- Applications and Benefits*', International Conference on Responsive Manufacturing (ICRM) 2002, June 26-29, 2002, Turkey

Luo, R.C., Tzou, Jyh H., and Chang, Y.C., (2001), '*An Internet-based Remote Control and Monitoring Rapid Prototyping System*', Industrial Electronics Society, The 27th Annual Conference of the IEEE, pp: 159-164 vol.1

Luo, R.C., Tzou, Jyh H., and Chang, Y.C., (2001), '*Desktop Rapid Prototyping System with Supervisory Control and Monitoring Through Internet*', Mechatronics, IEEE/ASME Transactions, pp: 399-409

Rapid-Prototyping Tutorial, http://home.att.net/~eastleisland/sls_int.htm

Rapid-Prototyping Tutorial, http://www.3trpd.co.uk/data-interchange.htm

Rapid-Prototyping Tutorial, http://www.cs.hut.fi/~ado/rp/section3_7.html#SECTION00070000000000000000

Rapid-Prototyping Tutorial, http://www.sdsc.edu/tmf/Stl-specs/stl.html

Remote PC Access home page: www.access-remote-pc.com

Shade, Bill, (2002), '*E-manufacturing Solutions Converge*', Semiconductor International; Newton; Jul 2002; Volume: 25, Issue:8 , pp: 222-223

Singer, Peter, (2001), '*The Power of e-Manufacturing*', Semiconductor International, Volume: 24, Issue: 8, pp: 19-21

Tan, Ki Ngiam, Kwong, K.C. Feh Tay, Tan, K.K., Toh, T.W., Goh, E.M., Khanal, E.M., (1999), '*Internet manufacturing (IMan)*', Department of Mechanical & Production Engineering, National University of Singapore; Internet Workshop. IWS 99 02/18/1999 - 02/20/1999, Osaka, Japan, pp: 69-74

Tay, Francis E.H., Khanal, Yadav P., Kwong, Kwok Kuen, and Tan, Kim Cheng, (2000), '*Distributed Rapid Prototyping - a Framework for Internet Prototyping and Manufacturing*', Integrated Manufacturing Systems 2001, Volume: 12 Number: 6 Page: 409 - 415

Trebilcock, Bob, (2001), '*E-manufacturing Gets Started*', Modern Materials Handling; Boston; Mid-May 2001; Page: 15-19

Waurzyniak, Patrick, (2001), '*Electronic Intelligence in Manufacturing*', Manufacturing Engineering, Vol. 127/No 4, pp: 44-67

Waurzyniak, Patrick, (2001), '*Moving Toward the e-Factory*', Manufacturing Engineering, Vol. 127/No 4, pp: 43-60

Waurzyniak, Patrick, (2001), '*Web Tools Catch On*', Manufacturing Engineering, Vol. 127/No 4, pp: 82-102

Weimer, George, (2000), '*E-manufacturing is Clicking on*', Material Handling Management, Volume:55, Issue:7, pp:28-30

ADDITIONAL REFERENCES

Alexander, Michael, (2001), '*Factory Floors Go Online - Pioneering Manufacturers Close the Final Gap in their Supply Chains*', Issue: 852, pp: 21-22

Anonymous, (2000), '*E-manufacturing: 'a long way to go*', Works Management, Dec 2000; pp: 5-6

Anonymous, (2000), '*Top 100: E-Manufacturing*', Manufacturing Systems, Volume: 18, Issue: 7, pp: 45-48

Anonymous, (2001), '*Case Studies: Simulation'*, Robotics World, ISSN: 07377908, pp: 31-32

Anonymous, (2001), '*E-Manufacturing: ready or not?'*, Manufacturing Engineer Society, pp: 38-39

Bloss, Dave, and Pillai, Devadas, (2001), '*E-Manufacturing Opportunities in Semiconductor Processing*', Semiconductor International; Volume: 24, Issue: 8, pp: 88-96

Bylinsky, G., (2000), '*New Hot Technologies for American Factories*', Industrial Management & Technology Section, Fortune Magazine, http://www.fortune.com

Collins J., (1999), '*Data Expression, a Gigabit Junction with the next Generation Internet*', IEEE Spectrum, pp: 31-42

GE Fanuc Website:
http://www.google.ca/search?q=cache:S9ddo8M3YDAJ:www.plantservices.com/Web_First/ps.nsf/ArticleID/DPIC569KCN/+Kevin+Roach,+eManufacturing&hl=en&lr=lang_en&ie=UTF-8

Gould, Lawrence S., (2000), '*Unigraphic goes interoperable*', Automotive Manufacturing & Production, Vol.: 112, Issue: 10, pp: 48-49

Hamel, William R., (2000), '*E-maintenance in Hazardous Environments*', IEEE/RSJ, pp: 838-842

Heimberg, Ralf, (2000), '*E-Manufcaturing*', Innovative Technologies, pp: 313-315

Heinicke, Matthias U., (2000), '*Eliminate Bottlenecks With Integrated Analysis Tools in eM-PLANT*', Winter Simulation Conference, pp: 229-231

Hickey, Katheleen, (2000), '*E-manufacturing*', Traffic World, Vol.: 263, Issue: 11, pp: 24-27

Internet Study Website of UBC- http://www.highlatitude.com/whiff_of_grape_nov00/sld015.htm

Java Company Website - http://java.sun.com/

Macdonald, Cindy, (2001), '*It's EDI all over again*', Canadian Plastics, ISSN: 084778, pp: 14

Macdonald, Cindy, (2001), '*The Wired Plant*', Canadian Plastics, Volume: 59, Issue: 2, pp: 29-42

Maloney, David, (2002), '*E-Manufacturing e'zing into Factories*', Modern Materials Handling; Volume: 57, Issue: 6, pp: 21-23

Michel, Roberto, (2000), '*E-manufacturing Essentials*', Manufacturing Systems, Volume: 18, Issue: 5, pp: 36-44

Michel, Roberto, (2001), '*E-manufacturing with an "I"*', Manufacturing Society, Volume: 19, Issue: 3, pp: 16-18

Montoya, Robert J., and Williams, Gary, (2000), '*Transforming Intel Factories from Global Present to Global Operation*', The Ninth International Symposium Semiconductor Manufacturing, pp: 17-20

Oracle Company Website- http://www.oracle.com

Oracle Tutorial Website - http://cimic.rutgers.edu/~holowcza/oracle/sqlplus/f841a885f83693e8bde7fd9c4fa38c41.html

Pham, D.T., and Gault, R.A., (1998), '*Comparison of Rapid Prototyping Technologies*', International Journal of Machine Tools and Manufacture, pp: 1257-1287

Rapid Prototyping Tutorial, http://www.cs.hut.fi/~ado/rp/section3_5.html

Roberts, Michael, (2000), '*Using e-Manufacturing to Drive Supply Chain Optimization*', Chemical Week, pp: 21-22

Shalvi, Marsha, (2001), '*E-manufacturing finally takes its Place in the Sun*', Machine Design, pp: 148-150

Sielinski, Ron, (2000), '*Microsoft in Manufacturing*', Manufacturing Engineering, Volume: 125, Issue: 1, pp: 71-82

Singer, Peter, (2002), '*Getting a grip on the E-manufacturing E-lephant*', Semiconductor International; pp: 11-14

STEP Tools Inc. Website - http://www.steptools.com/library/cts_99/sld011.htm

Taylor, K., and Dalton, B., (2000), '*Internet Robots: A New Robotics Niche*', IEEE Robotics and Automation Magazine, Volume: 7, Issue: 1

Teresko, John, (2000), '*The Dawn of e-Manfacturing*', Industry Week, Vol.: 249, Issue: 16, pp: 55-60

Wiersema, William H., and Jones, Kevin, (2000), '*E-commerce: Opportunity or fad?*', Electrical Apparatus; Volume: 53, Issue: 7, pp: 46-47

Wraige, Helen, (2001), '*Right from the Start*', Professional Engineering, Bury St. Edmunds; Mar 28, 2001; Volume: 14, Issue: 6, pp: 41-43

Zurawski, Laura, (2001), '*E-manufacturing a Catchy Name for What You Should be Doing Anyway*', Control Engineering, pp: 79-86

APPENDIX

Appendix A- Codes for e-Manufacturing Data-Exchange System

(A) JSP codes

1. an.jsp-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>Remote Monitering</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td>Hello
</td>
<td>Hello
</td>
</tr></table>
<p>Hello, <%=session.getAttribute("UserID")%></p>
<p>Current time: <%=new java.util.Date()%></p>
<p><b>File download.</b></p>
<p>
```

```
<table>
 <tr>
  <td>
   Surface cutting video:
  </td>
  <td>
   <a href="030430-001.avi">Video-1</a>
  </td>
  <td>
   <a href="030430-002.mpeg">Video-2</a>
  </td>

 </tr>
 <tr>
  <td>
   CNC programs:
  </td>
  <td>
   <a href="part001.nc">Part001</a>
  </td>
  <td>
   <a href="part002.nc">Part002</a>
  </td>

 </tr>
</table>
<p>
<p><b>File upload.</b></p>
<p>
<form name="upload1" action="an.jsp" enctype="multipart/form-data"
method="post">
  Enter data file below:<br>
  <input type="file" name="filename">
```

```
  </form>
</p>
<p>If you want to go to remote controlling system:</p>
<p>
<form method="POST">
  <table>
   <tr>
    <td>
     Click to remote controlling system:
    </td>
    <td>
     <input type="submit" value="Remote controlling system">
    </td>
   </tr>
  </table>
  </form>
</p></p>If you want to log out:</p>
<p>
<form method="POST" action="logoutServlet">
  <table>
   <tr>
    <td>
     Click to log out:
    </td>
    <td>
     <input type="submit" value="Log out">
    </td>
   </tr>
  </table>
  </form>
</p>
```

```
</body>
</html>
```

2. me.jsp-

```
<DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>Me</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href="me.jsp">Supplier</a>
</td>
<td><a href="mes.jsp">Client</a>
</td>
</tr></table>
<p>Hello, <%=session.getAttribute("UserID")%></p>
<p>Current time: <%=new java.util.Date()%></p>
<p>Please comfirm receiving.</p>
<p>
<form method="POST" action="merServlet">
  <table>
    <tr>
```

```
      <td>
       Lot Number(i.e. 1002):
      <td>
      <td>
       <input type="text" name="LotNum">
      <td>
     <tr>
     <tr>
      <td>
       Click to confirm receiving
      <td>
      <td>
       <input type="submit" value="Submit">
      <td>
     <tr>
   <table>
   <form>
<p>
<p>Search information.<p>
<p>
<form method="POST" action="rmsServlet">
   <table>
    <tr>
     <td>
      Lot Number(i.e. 1002):
     <td>
     <td>
      <input type="text" name="LotNum">
     <td>
    <tr>

    <tr>
     <td>
```

```
        Click to search:
      <td>
      <td>
        <input type="submit" value="Search">
      <td>
    <tr>
  <table>
  <form>
<p>
<p>If you want to log out:<p>
<p>
<form method="POST" action="logoutServlet">
  <table>
    <tr>
      <td>
        Click to log out:
      <td>
      <td>
        <input type="submit" value="Log out">
      <td>
    <tr>
  <table>
  <form>
<p>

<body>
<html>
```

3. mes.jsp-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>My product</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href="me.jsp">Supplier</a>
</td>
<td><a HREF="mes.jsp">Client</a>
</td>
</tr></table>
<p>Hello, <%=session.getAttribute("UserID")%></p>
<p>Current time: <%=new java.util.Date()%></p>
<p>Please submit or correct information.</p>
<p>
<form method="POST" action="shipServlet">
  <table>
   <tr>
    <td>
     Lot Number(i.e. 10002):
    </td>
    <td>
     <input type="text" name="LotNum">
    </td>
   </tr>
   <tr>
```

```
      <td>
       Row material lot number(i.e. 1002/3):
      </td>
      <td>
       <input type="text" name="RMLN">
      </td>
     </tr>
     <tr>
      <td>
       Ship Date(YY/MM/DD):
      </td>
      <td>
       <input type="text" name="ShipDate">
      </td>
     </tr>
     <tr>
      <td>
       Click to submit:
      </td>
      <td>
       <input type="submit" value="Submit">
      </td>
     </tr>
    </table>
   </form>
</p>
<p>Search product information.</p>
<p>
<form method="POST" action="piServlet">
   <table>
    <tr>
     <td>
      Lot Number(i.e. 10002):
```

```
    </td>
    <td>
      <input type="text" name="LotNum">
    </td>
  </tr>

  <tr>
    <td>
      Click to search:
    </td>
    <td>
      <input type="submit" value="Search">
    </td>
  </tr>
 </table>
 </form>
</p>
<p>Delete product information.</p>
<p>
<form method="POST" action="pdServlet">
  <table>
    <tr>
      <td>
        Lot Number(i.e. 10002):
      </td>
      <td>
        <input type="text" name="LotNum">
      </td>
    </tr>

    <tr>
      <td>
        Click to delete:
```

```
        <td>
        <td>
          <input type="submit" value="Delete">
        <td>
       <tr>
     <table>
     <form>
  <p>

  <body>
  <html>
```

4. pixml.jsp-

```
<?xml version ="1.0" ?>
<%@ page
import="java.sql.*,java.io.*,java.util.*,javax.servlet.*,javax.servlet.http.*"
%>
<%
String lotNum=(String)session.getAttribute("LotNum");



String rString=lotNum+",";
   Connection connection;
   Statement statement;
   ResultSet resultSet;
   Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");


   String query;
```

```
query="select shipdate from pt where lotnum='"+lotNum+"'";

statement=connection.createStatement();
resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
 query="select rmln from pt where lotnum='"+lotNum+"'";
 resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
   query="select status from pt where lotnum='"+lotNum+"'";
 resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
resultSet.close();
statement.close();
connection.close();
 StringTokenizer tokens=new StringTokenizer(rString, ", ");
      try{
      out.println("<lot>"+"\n"

       +"<lotNumber>"+tokens.nextToken()+"</lotNumber>"+"\n"
            +"<ShipDate>"+tokens.nextToken()+"</ShipDate>"+"\n"

       +"<RawMaterialLotNumber>"+tokens.nextToken()+"</RawMater
ialLotNumber>"+"\n"
            +"<Status>"+tokens.nextToken()+"</Status>"+"\n"
            +"</lot>"+"\n");}catch(IOException e){}
%>
```

5. rm.jsp-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>Supplier</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href='rm.jsp">Submit</a>
</td>
<td><a href="rms.jsp">Search</a>
</td>
</tr></table>
<p>Hello, <%=session.getAttribute("UserID")%></p>
<p>Current time: <%=new java.util.Date()%></p>
<p>Please submit or correct information.</p>
<p>
<form method="POST" action="rmServlet">
  <table>
    <tr>
      <td>
        Lot Number(i.e. 1002):
```

```
    </td>
    <td>
      <input type="text" name="LotNum">
    </td>
  </tr>
  <tr>
    <td>
      Hardness(HB/HC):
    </td>
    <td>
      <input type="text" name="Hardness">
    </td>
  </tr>
  <tr>
    <td>
      Ship Date(YY/MM/DD):
    </td>
    <td>
      <input type="text" name="ShipDate">
    </td>
  </tr>
  <tr>
    <td>
      Click to submit:
    </td>
    <td>
      <input type="submit" value="Submit">
    </td>
  </tr>
</table>
</form>
<p>
<p>If you want to log out:</p>
```

```
<p>
<form method="POST" action="logoutServlet">
  <table>
   <tr>
    <td>
      Click to log out:
    </td>
    <td>
      <input type="submit" value="Log out">
    </td>
   </tr>
  </table>
  </form>
</p>

</body>
</html>
```

6. rms.jsp-

```
< DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>Supplier</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
```

```
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href="rm.jsp">Submit</a>
</td>
<td><a href="rms.jsp">Search</a>
</td>
</tr></table>
<p>Hello, <%=session.getAttribute("UserID")%></p>
<p>Current time: <%=new java.util.Date()%></p><p>Search
information.</p>
<p>
<form method="POST" action="rmsServlet">
  <table>
   <tr>
    <td>
     Lot Number(i.e. 1002):
    </td>
    <td>
     <input type="text" name="LotNum">
    </td>
   </tr>

   <tr>
    <td>
     Click to search:
    </td>
    <td>
     <input type="submit" value="Search">
    </td>
   </tr>
  </table>
  </form>
</p>
<p>Delete information.</p>
```

```
<p>
<form method="POST" action="rmdServlet">
  <table>
   <tr>
    <td>
     Lot Number(i.e. 1002):
    </td>
    <td>
     <input type="text" name="LotNum">
    </td>
   </tr>

   <tr>
    <td>
     Click to delete:
    </td>
    <td>
     <input type="submit" value="Delete">
    </td>
   </tr>
  </table>
  </form>
</p>

</body>
</html>
```

7. srchxml.jsp-

```
<?xml version ="1.0" ?>
```

```
<%@ page
import="java.sql.*,java.io.*,java.util.*,javax.servlet.*,javax.servlet.http.*"
%>
<%      String aString=(String)session.getAttribute("Srch");
    StringTokenizer tokens=new StringTokenizer(aString, ", ");
        try{
        out.println("<lot>"+"\n"

        +"<lotNumber>"+tokens.nextToken()+"</lotNumber>"+"\n"
                +"<ShipDate>"+tokens.nextToken()+"</ShipDate>"+"\n"
                +"<Hardness>"+tokens.nextToken()+"</Hardness>"+"\n"
                +"<Status>"+tokens.nextToken()+"</Status>"+"\n"
                +"</lot>"+"\n");}catch(IOException e){}
%>
```

(B) HTML codes

1. emanu.html-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>E-Manufactory</title>
</head>
<body>
<p><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br />
```

```
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><h
href="eserv.html">E-Service</a>
</td>
<td><a href="emanu.html">E-Manufactory</a>
</td>
</tr></table>
<p>Please log in.</p>
<form method="POST" action="loginServlet">
  <table>
   <tr>
    <td>
     User ID:
    </td>
    <td>
     <input type="text" name="UserID">
    </td>
   </tr>
   <tr>
    <td>
     Password:
    </td>
    <td>
     <input type="password" name="Password">
    </td>
   </tr>
   <tr>
    <td>
     Click to log in:
    </td>
    <td>
     <input type="submit" value="Log In">
```

```
    </td>
   </tr>
  </table>
  </form>
</body>
</html>
```

2. emanu2.html-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>E-Manufactory</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Manufactory</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href="eserv.html">E-Service</a>
</td>
<td><a href="emanu.html">E-Manufactory</a>
</td>
</tr></table>
<p>Log in incorrect. Please check your userID and password, then try
again.</p>
<form method="POST" action="loginServlet">
  <table>
   <tr>
    <td>
     User ID:
```

```
    </td>
    <td>
      <input type="text" name="UserID">
    </td>
  </tr>
  <tr>
    <td>
      Password:
    </td>
    <td>
      <input type="password" name="Password">
    </td>
  </tr>
  <tr>
    <td>
      Click to log in:
    </td>
    <td>
      <input type="submit" value="Log In">
    </td>
  </tr>
 </table>
 </form>
</body>
</html>
```

3. eserv.html-

```
<! DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<titil>E-Service</titil>
```

```
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>E-Service</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
<table width="100%" bgcolor="#ffcc00"><tr><td><a
href="eserv.html">E-Service</a>
</td>
<td><a href="emanu.html">E-Manufactory</a>
</td>
</tr></table>
<p>
Under construction. It will come soon.
</p>
</body>
</html>
```

4. index.html-

```
<! DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>My Project</title>
</head>
<body>
<P><img src="logo.gif" alt="M&A Logo" align="left" /><h1><font
color="#ffcc00"><b>My Project</b></font></h1></p>
<br/>
<hr size=5 color="#990000" />
<br/>
```

<table width="100%" bgcolor="#ffcc00" ><tr><td ><a
href="eserv.html" >E-Service

</td >

<td ><a href="emanu.html" >E-Manufactory

</td >

</tr></table >

Through centuries, the world has witnessed a lot of changes. One of the most remarkable changes

is that people use internet as a new market place.

<P >I always consider if we can use same technology at after sale service and manufactoring.

Therefore, Enterprises can offer more effective and efficient services and reduce the costs.

So I try <a href="eserv.html" >E-Service and <a
href="emanu.html" >E-Manufactory systems.

</body >

</html >

(C) XML Codes

Web.xml-

<?xml version="1.0" encoding="ISO-8859-1"? >

<!DOCTYPE web-app

PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"

```
"http://java.sun.com/j2ee/dtds/web-app_2_2.dtd" >
<web-app >
  <servlet >
    <servlet-name >loginServlet </servlet-name >
    <servlet-class >loginServlet </servlet-class >
  </servlet >
  <servlet-mapping >
        <servlet-name >loginServlet </servlet-name >
        <url-pattern >/loginServlet </url-pattern >
  </servlet-mapping >
  <servlet >
    <servlet-name >rmServlet </servlet-name >
    <servlet-class >rmServlet </servlet-class >
  </servlet >
  <servlet-mapping >
        <servlet-name >rmServlet </servlet-name >
        <url-pattern >/rmServlet </url-pattern >
  </servlet-mapping >
  <servlet >
    <servlet-name >logoutServlet </servlet-name >
    <servlet-class >logoutServlet </servlet-class >
  </servlet >
  <servlet-mapping >
        <servlet-name >logoutServlet </servlet-name >
        <url-pattern >/logoutServlet </url-pattern >
  </servlet-mapping >
  <servlet >
    <servlet-name >rmdServlet </servlet-name >
    <servlet-class >rmdServlet </servlet-class >
  </servlet >
  <servlet-mapping >
        <servlet-name >rmdServlet </servlet-name >
        <url-pattern >/rmdServlet </url-pattern >
```

</servlet-mapping>
<servlet>
  <servlet-name>rmsServlet</servlet-name>
  <servlet-class>rmsServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>rmsServlet</servlet-name>
    <url-pattern>/rmsServlet</url-pattern>
</servlet-mapping>
<servlet>
  <servlet-name>merServlet</servlet-name>
  <servlet-class>merServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>merServlet</servlet-name>
    <url-pattern>/merServlet</url-pattern>
</servlet-mapping>
<servlet>
  <servlet-name>shipServlet</servlet-name>
  <servlet-class>shipServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>shipServlet</servlet-name>
    <url-pattern>/shipServlet</url-pattern>
</servlet-mapping>
<servlet>
  <servlet-name>pdServlet</servlet-name>
  <servlet-class>pdServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>pdServlet</servlet-name>
    <url-pattern>/pdServlet</url-pattern>
</servlet-mapping>

```
<servlet>
  <servlet-name>piServlet</servlet-name>
  <servlet-class>piServlet</servlet-class>
</servlet>
<servlet-mapping>
      <servlet-name>piServlet</servlet-name>
      <url-pattern>/piServlet</url-pattern>
</servlet-mapping>
</web-app>
```

(D) JAVA Codes

1. accdb.java-

```
/* This class is for data base access, logint is log in table, rmt is raw
material table and pt is product table */


import java.sql.*;
import java.util.*;
import java.awt.*;

public class Accdb{


 public Accdb(){}


 /* This method is used to check user name and password and return who
it is */
 public String login(String userid, String password) throws
SQLException, Exception{
   String rString="";
   Connection connection;
```

```
Statement statement;
ResultSet resultSet;
Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

String query;

query="select WHO from logint where USERID='"+userid+"' AND
PASSWORD='"+password+"'";

statement=connection.createStatement();
resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1);}
resultSet.close();
statement.close();
connection.close();
return rString;
}
/* This method is used to search raw material information */
public String rms(String lotNum) throws SQLException, Exception{
String rString="";
Connection connection;
Statement statement;
ResultSet resultSet;
Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

String query;
```

```
query="select shipdate from rmt where lotnum='"+lotNum+"'";

statement=connection.createStatement();
resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
 query="select hardness from rmt where lotnum='"+lotNum+"'";
 resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
   query="select status from rmt where lotnum='"+lotNum+"'";
 resultSet=statement.executeQuery(query);
while(resultSet.next())
{rString+=resultSet.getString(1)+",";
   }
resultSet.close();
statement.close();
connection.close();
return rString;

}
/* This method is used to search product information */
public String productI(String lotNum) throws SQLException,
Exception{
String rString="";
Connection connection;
Statement statement;
ResultSet resultSet;
Class.forName("oracle.jdbc.driver.OracleDriver");
```

```
connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

   String query;


   query="select shipdate from pt where lotnum='"+lotNum+"'";

   statement=connection.createStatement();
   resultSet=statement.executeQuery(query);
   while(resultSet.next())
   {rString+=resultSet.getString(1)+",";
     }
    query="select rmln from pt where lotnum='"+lotNum+"'";
    resultSet=statement.executeQuery(query);
   while(resultSet.next())
   {rString+=resultSet.getString(1)+",";
     }
     query="select status from pt where lotnum='"+lotNum+"'";
    resultSet=statement.executeQuery(query);
   while(resultSet.next())
   {rString+=resultSet.getString(1)+",";
     }
   resultSet.close();
   statement.close();
   connection.close();

   return rString;

   }
   /* This method is used to add raw material ship information */
```

```
   public void rmsubmit(String lotNum, String hardness, String shipDate)
throws SQLException, Exception{

   Connection connection;
   Statement statement;
   ResultSet resultSet;
   Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

   String query;


   query="insert into rmt values('"+lotNum+"',
'"+shipDate+"','"+hardness+"','shipped')";

   statement=connection.createStatement();
   resultSet=statement.executeQuery(query);
   resultSet.close();
   statement.close();
   connection.close();

   }
   /* This method is used to add product ship information */
   public void shipP(String lotNum, String rmln, String shipDate) throws
SQLException, Exception{

   Connection connection;
   Statement statement;
   ResultSet resultSet;
   Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");
```

```
    String query;


    query="insert into pt values('"+lotNum+"',
'"+shipDate+"','"+rmln+"','shipped')";

    statement=connection.createStatement();
    resultSet=statement.executeQuery(query);
    resultSet.close();
    statement.close();
    connection.close();

    }
    /* This method is used to delete raw material information */
    public void rmd(String lotNum) throws SQLException, Exception{

    Connection connection;
    Statement statement;
    ResultSet resultSet;
    Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

    String query;


    query="delete from rmt where lotnum='"+lotNum+"'";

    statement=connection.createStatement();
    resultSet=statement.executeQuery(query);
    resultSet.close();
```

```
    statement.close();
    connection.close();

    }
    /* This method is used to delete product information */
    public void productD(String lotNum) throws SQLException,
Exception{

    Connection connection;
    Statement statement;
    ResultSet resultSet;
    Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

    String query;


    query="delete from pt where lotnum='"+lotNum+"'";

    statement=connection.createStatement();
    resultSet=statement.executeQuery(query);
    resultSet.close();
    statement.close();
    connection.close();

    }
    /* This method is used to indicate receiving of raw material  */
    public void mer(String lotNum) throws SQLException, Exception{

    Connection connection;
    Statement statement;
```

```
ResultSet resultSet;
Class.forName("oracle.jdbc.driver.OracleDriver");

connection=DriverManager.getConnection("jdbc:oracle:thin:@goedel.new
cs.uwindsor.ca:1521:CS01","xie6","xie8xu");

String query;

query="update rmt set status='received' where lotnum='"+lotNum+"'";

statement=connection.createStatement();
resultSet=statement.executeQuery(query);
resultSet.close();
statement.close();
connection.close();

}

}
```

2. loginservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class loginServlet extends HttpServlet {
```

```
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
        HttpSession session =request.getSession(true);
        response.setContentType("text/html");
        try{PrintWriter out =response.getWriter();}catch (IOException e)
{}

//get username and password
        String userID =request.getParameter("UserID");
        String password =request.getParameter("Password");
        session.setAttribute("UserID", userID);
//confirm user name and password, get who it is and lead to related pages

        Accdb aAccdb=new Accdb();
        String aString="";
              try{
        aString+=aAccdb.login(userID, password);}catch(SQLException
e){}catch(Exception e1){}

        if(aString.equals("rm")){
        try {
              response.sendRedirect("rm.jsp");
        }catch (IOException e) {}}
        else if(aString.equals("me")){
        try {
              response.sendRedirect("me.jsp");
        }catch (IOException e) {}}

    else if(aString.equals("an")){
        try {
              response.sendRedirect("an.jsp");
        }catch (IOException e) {}}
    else {
        try {
```

```
                response.sendRedirect("emanu2.html");
            }catch (IOException e) {}

        }
    }

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

logoutservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class logoutServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
     HttpSession session=request.getSession(true);
        //close session
        session.invalidate();
        response.setContentType("text/html");
        try{PrintWriter out =response.getWriter();}catch (IOException e)
{}

//go back to index page
```

```
        try {
                response.sendRedirect("index.html");
        }catch (IOException e) {}



}

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

merservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class merServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {



        String lotNum =request.getParameter("LotNum");
        //confirm receiving of raw material
        if (!lotNum.equals(""))         {
        Accdb aAccdb=new Accdb();
                try{
```

```
        aAccdb.mer(lotNum);}catch(SQLException e){}catch(Exception
e1){}
        }

        try {
                response.sendRedirect("me.jsp");
        }catch (IOException e) {}

 }

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

3. pdservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class pdServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {


        String lotNum =request.getParameter("LotNum");
        //delete product information
```

```
        if (!lotNum.equals(""))         {
        Accdb aAccdb =new Accdb();
                try{
        aAccdb.productD(lotNum);}catch(SQLException
e){}catch(Exception e1){}
        }

        try {
                response.sendRedirect("mes.jsp");
        }catch (IOException e) {}

}

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

4. piservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class piServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
        HttpSession session =request.getSession(true);
```

```
        String lotNum =request.getParameter("LotNum");

        response.setContentType("text/html");
        //transfer lot num to be used at pixml.jsp
        session.setAttribute("LotNum", lotNum);
        //redirect to a jsp to generate xml to display product information
        try {
                response.sendRedirect("pixml.jsp");
        }catch (IOException e) {}



  }

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

rmdservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class rmdServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
```

```
        String lotNum =request.getParameter("LotNum");
        //delete raw material information
        if (!lotNum.equals(""))         {
        Accdb aAccdb=new Accdb();
                try{
        aAccdb.rmd(lotNum);}catch(SQLException e){}catch(Exception
e1){}
        }

        try {
                response.sendRedirect("rms.jsp");
        }catch (IOException e) {}

}

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

rmservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
```

```
public class rmServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {


        String lotNum =request.getParameter("LotNum");
        String hardness =request.getParameter("Hardness");
        String shipDate =request.getParameter("ShipDate");
        //submit raw material ship information to data base
        if (!lotNum.equals(""))         {
        Accdb aAccdb=new Accdb();
                try{
        aAccdb.rmsubmit(lotNum, hardness,
shipDate);}catch(SQLException e){}catch(Exception e1){}
        }

        try {
                response.sendRedirect("rm.jsp");
        }catch (IOException e) {}

}

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

rmsservlet.java-

```
import java.sql.*;
import java.util.*;
```

```
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class rmsServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
        HttpSession session =request.getSession(true);
        //get raw material information from data base
     Accdb aAccdb=new Accdb();
        //sent information to be used at srchxml.jsp to generate raw
material information
        String lotNum =request.getParameter("LotNum");

        response.setContentType("text/html");



                String aString=lotNum+",";

                try{
        aString+=aAccdb.rms(lotNum);}catch(SQLException
e){}catch(Exception e1){}
        session.setAttribute("Srch", aString);
        try {
                response.sendRedirect("srchxml.jsp");
        }catch (IOException e) {}
/*      StringTokenizer tokens=new StringTokenizer(aString, ", ");
```

```
        try{
        PrintWriter out =response.getWriter();

        out.println(docType+"\n"
                +"<lot>"+"\n"
                +"<lotNumber>"+lotNum+"</lotNumber>"+"\n"
                +"<ShipDate>"+tokens.nextToken()+"</ShipDate>"+"\n"
                +"<Hardness>"+tokens.nextToken()+"</Hardness>"+"\n"
                +"<Status>"+tokens.nextToken()+"</Status>"+"\n"
                +"</lot>"+"\n");}catch(IOException e){}*/



        }

  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

shipservlet.java-

```
import java.sql.*;
import java.util.*;
import java.awt.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class shipServlet extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse
response) {
```

```
        String lotNum =request.getParameter("LotNum");
        String rmln =request.getParameter("RMLN");
        String shipDate =request.getParameter("ShipDate");
        //submit product ship information

        if (!lotNum.equals(""))         {
        Accdb aAccdb=new Accdb();
                try{
        aAccdb.shipP(lotNum, rmln, shipDate);}catch(SQLException
e){}catch(Exception e1){}
        }
        try {
                response.sendRedirect("mes.jsp");
        }catch (IOException e) {}


}
  public void doPost(HttpServletRequest request, HttpServletResponse
response) {
        doGet(request, response);
  }
}
```

Appendix B- User Guide for e-Manufacturing Data-Exchange System

1. First requirement for the system is a computer with Internet access.
2. Turn on the Sever for system page, for example, in this demonstration, we are using the Tomcat server, so go to tomcat bin by typing: **cd cs334/tomcat4/bin,** then start the server with a command: **startup 47026**
3. In Internet Browser, open the system page by typing in the address for system page: http://davinci.newcs.uwindsor.ca:47026/myproject/index.html
4. In the face page, the button of 'e-Manufactory' is for entering the created Manufacturing Data Exchange system, while 'e-Service' button is still under construction, it is reserved for the development of e-prediction or e-Maintenance purpose.
5. By clicking the button of 'e-Manufactory', user name and password are then required for safety issue. Different user will be assigned different user name/password, which will lead to specific web pages. For example, in the demonstration, the assignments are as listed below:

Table 4 The assignment of user names in the created system

User names	**Assigned to**	**Page contents**
marvin	Manufacturer	Data input to customer/Search/Delete data
aacast	Raw Material Supplier	Input/Search data to manufacturer
angela	Sister Company	Download/Upload files

6. Web page functionality is as buttons indicated at the pages, including Data submitting, search/ Delete/ Download/ Upload
7. Shut down the server can be done with a command: **shutdown_tomcat**

Appendix C- The Output SML file of Remote Utilization of Catalyst Software

```
# Build style -  November 07, 2003 12:02:40 ;
# Written by Catalyst
# Product version 2.1
# Build version 1570
setVersionVar modelerType mariner;
setVersionVar modelMaterial "P400";
setVersionVar modelTip T12;
setVersionVar supportMaterial "P400_r";
setVersionVar supportTip T12;
setVersionVar sliceHeight 0.013;

confirmModelerSetup dontCheck;


# Create custom groups, if any


global restoreLock;
if { $restoreLock != "restoreSets" } {
        errormsg "Cannot restore a set file without restoreSets.";
        return;
};
                                # Variables in sslParams
setVersionVar addBase           {true} 1.0;
setVersionVar allSliceBBox  { 1.682682 1.703104 1.123255 } 1.0;
setVersionVar alt:base:perimeterWidth       {0.026} 1.0;
setVersionVar alt:base:rasterAir        {0.065} 1.0;
setVersionVar alt:base:topAir           {-0.0018} 1.0;
setVersionVar alt:base:topWidth         {0.018} 1.0;
```

```
setVersionVar alt:base:width {0.026} 1.0;
setVersionVar alt:baseLevels {4} 1.0;
setVersionVar alt:baseOversize {0.050} 1.0;
setVersionVar alt:firstLayer:width {0.030} 1.0;
setVersionVar alt:firstLayer:z {0.013} 1.0;
setVersionVar alt:material:closedExitAngle {0} 1.0;
setVersionVar alt:material:closedExitDist {0.1} 1.0;
setVersionVar alt:material:closedExitEdge {startedge} 1.0;
setVersionVar alt:material:nPartXCap {1} 1.0;
setVersionVar alt:material:openExitAngle {180} 1.0;
setVersionVar alt:material:openExitDist {0.1} 1.0;
setVersionVar alt:material:type {P400_r} 1.0;
setVersionVar alt:rasterPen:constantFactor {0.08} 1.0;
setVersionVar alt:support:faceOffset {0.035} 1.0;
setVersionVar alt:support:growAmount {0.001} 1.0;
setVersionVar alt:support:growLimit {0.1} 1.0;
setVersionVar alt:support:perforate {false} 1.0;
setVersionVar alt:support:perforateHeight {1.0} 1.0;
setVersionVar alt:support:prefContourWidth {0.015} 1.0;
setVersionVar alt:support:selfSupport {45.0} 1.0;
setVersionVar alt:support:sparseAge {5} 1.0;
setVersionVar alt:support:sparsePerimGap {0.015} 1.0;
setVersionVar alt:support:sparseRasterAir {0.250} 1.0;
setVersionVar alt:support:sparseRasterWidth {0.020} 1.0;
setVersionVar alt:support:supPerfPerimGap {0.003} 1.0;
setVersionVar alt:support:supfaceAir {-0.0018} 1.0;
setVersionVar alt:support:supfacePerimGap {0.005} 1.0;
setVersionVar alt:support:supfaceWidth {0.018} 1.0;
setVersionVar alt:support:supportAir {0.0455} 1.0;
setVersionVar alt:support:supportPerimGap {0.010} 1.0;
setVersionVar alt:support:supportWidth {0.015} 1.0;
setVersionVar altMatlVolume {0.149} 1.0;
setVersionVar bottomZ {0.000000} 1.0;
```

```
setVersionVar currentLayerNo        {5} 1.0;
setVersionVar currentLayerZ         {0.013000} 1.0;
setVersionVar estimatedTime         {4247} 2.0;
setVersionVar fillMode      {normal} 1.0;
setVersionVar growSupports {smallgrow} 1.0;
setVersionVar main:base:perimeterWidth    {0.026} 1.0;
setVersionVar main:base:rasterAir   {0.065} 1.0;
setVersionVar main:base:topAir      {-0.0018} 1.0;
setVersionVar main:base:topWidth    {0.018} 1.0;
setVersionVar main:base:width       {0.026} 1.0;
setVersionVar main:baseLevels       {4} 1.0;
setVersionVar main:baseOversize     {0.050} 1.0;
setVersionVar main:firstLayer:width         {0.030} 1.0;
setVersionVar main:firstLayer:z     {0.013} 1.0;
setVersionVar main:material:closedExitAngle       {0} 1.0;
setVersionVar main:material:closedExitDist {0.1} 1.0;
setVersionVar main:material:closedExitEdge        {startedge} 1.0;
setVersionVar main:material:nPartXCap     {0} 1.0;
setVersionVar main:material:openExitAngle         {180} 1.0;
setVersionVar main:material:openExitDist  {0.1} 1.0;
setVersionVar main:material:type     {P400} 1.0;
setVersionVar main:part:CRAirGap  {0.000} 1.0;
setVersionVar main:part:contourDepth        {0.026} 1.2;
setVersionVar main:part:interiorRasterWidth       {0.013} 1.0;
setVersionVar main:part:prefContourWidth {0.026} 1.2;
setVersionVar main:part:rasterAngle         {45.0} 1.0;
setVersionVar main:part:rasterAngle2        {90.0} 1.0;
setVersionVar main:part:rasterWidth         {0.02} 1.0;
setVersionVar main:part:sparseAirGap        {0.15} 1.0;
setVersionVar main:part:sparseRasterDelta  {90.0} 1.0;
setVersionVar main:part:sparseRasterWidth {0.015} 1.0;
setVersionVar main:rasterPen:constantFactor       {0.08} 1.0;
setVersionVar main:rasterPen:useAngle       {false} 1.0;
```

```
setVersionVar main:rasterPen:useLength      {false} 1.0;
setVersionVar main:support:faceOffset       {0.035} 1.0;
setVersionVar main:support:growAmount       {0.001} 1.0;
setVersionVar main:support:growLimit        {0.1} 1.0;
setVersionVar main:support:perforate        {false} 1.0;
setVersionVar main:support:perforateHeight          {1.0} 1.0;
setVersionVar main:support:prefContourWidth         {0.015} 1.0;
setVersionVar main:support:selfSupport      {45.0} 1.0;
setVersionVar main:support:sparseAge        {5} 1.0;
setVersionVar main:support:sparsePerimGap           {0.015} 1.0;
setVersionVar main:support:sparseRasterAir          {0.250} 1.0;
setVersionVar main:support:sparseRasterWidth        {0.020} 1.0;
setVersionVar main:support:supPerfPerimGap          {0.003} 1.0;
setVersionVar main:support:supfaceAir       {-0.0018} 1.0;
setVersionVar main:support:supfacePerimGap          {0.005} 1.0;
setVersionVar main:support:supfaceWidth     {0.018} 1.0;
setVersionVar main:support:supportAir       {0.0455} 1.0;
setVersionVar main:support:supportPerimGap          {0.010} 1.0;
setVersionVar main:support:supportWidth     {0.015} 1.0;
setVersionVar mainMatlVolume        {0.686} 1.0;
setVersionVar oldToolpathMethod     {false} 1.0;
setVersionVar optimizeBase  {true} 1.0;
setVersionVar partDensity       {sparse} 1.0;
setVersionVar partFillStyle     {rasters} 1.0;
setVersionVar partQuality       {economy} 1.0;
setVersionVar partSliceBBox             { 1.553150 1.553150 1.066000 } 1.0;
setVersionVar reqSliceBottom            {0.000000} 1.0;
setVersionVar reqSliceTop       {12.000000} 1.0;
setVersionVar savedViews    {{jobSaveCurrent {0.435480 -0.154429 0.331175
0.000000 0.365411 0.184042 -0.394679 0.000000 0.000000 0.515216 0.240249
0.000000  -0.621951 -0.294600 -0.077336 1.000000}}} 1.0;
setVersionVar seamPlacement             {align} 1.0;
setVersionVar seamRefPoint { 0.0 0.0 } 1.0;
```

```
setVersionVar shadeToolpath          {false} 1.0;
setVersionVar sliceAutoClose         {true} 1.0;
setVersionVar sliceBottomZ {-0.039000} 1.0;
setVersionVar sliceFilterTolerance   {0.0008} 1.0;
setVersionVar sliceOpenTolerance   {0.01} 1.0;
setVersionVar sliceTopZ      {1.066000} 1.0;
setVersionVar stepCriteria     {any} 1.0;
setVersionVar stepGroup      {} 1.0;
setVersionVar stepIncrement {1} 1.0;
setVersionVar stlColor         {10} 1.0;
setVersionVar stlCullBackface        {true} 1.0;
setVersionVar stlOffset        { 0.0 0.0 } 1.0;
setVersionVar stlReverseNormals     {false} 1.0;
setVersionVar stlRotateIncrement     {90.0} 1.0;
setVersionVar stlRotateX      {0.0} 1.0;
setVersionVar stlRotateY      {0.0} 1.0;
setVersionVar stlRotateZ      {0.0} 1.0;
setVersionVar stlScale         {1.0} 1.0;
setVersionVar stlShadeMode {flat} 1.0;
setVersionVar stlSize {1.553100 1.553100 1.054300} 1.0;
setVersionVar stlUnits         {mm} 1.0;
setVersionVar styleVer         {1.0} 1.0;
setVersionVar supportBaseMode      {supportsAndBase} 1.0;
setVersionVar supportTop     {1.066000} 1.0;
setVersionVar supportType    {sparse} 1.0;
setVersionVar surroundDepth          {0.05} 1.0;
setVersionVar surroundGap   {0.005} 1.0;
setVersionVar surroundPart   {false} 1.0;
setVersionVar topZ    {1.054300} 1.0;
setVersionVar useSliceBottom         {false} 1.0;
setVersionVar useSliceTop    {false} 1.0;
setVersionVar useSupportTop          {false} 1.0;
setVersionVar userViews      {} 1.0;
```

```
setVersionVar viewLayerBelow      {false} 1.0;
setVersionVar viewRangeBottom     {0.0} 1.0;
setVersionVar viewRangeTop        {0.0} 1.0;
setVersionVar viewSTL      {false} 1.0;
setVersionVar viewSTLOptions      {shaded} 1.0;
setVersionVar viewTheseLayers     {current} 1.0;

null " "
view restore jobSaveCurrent
view delete jobSaveCurrent
```

Appendix D- The Output NC file of Remote Utilization of MasterCAM8.0

```
%
O0000
(PROGRAM NAME - SAMPLECUT3)
(DATE=DD-MM-YY - 28-04-03 TIME=HH:MM - 14:25)
N100G20
N102G0G17G40G49G80G90
( 7/16 BULL ENDMILL 0.0313 RAD TOOL - 1 DIA. OFF. - 41 LEN. - 1 DIA. - .4375)
N104T1M6
N106G0G90G54X.0001Y-1.434Z2.6953A0.S1222M3
N108G43H1Z2.6953
N110G1Z2.4453F8.41
N112X-.0043Y-1.4336
N114X-.0637Y-1.4327
N116X-.1271Y-1.431
N118X-.1928Y-1.4279
N120X-.2279Y-1.4255
N122X-.262Y-1.4226
N124X-.2965Y-1.4191
N126X-.3312Y-1.4149
N128X-.3662Y-1.41
N130X-.4014Y-1.4042
N132X-.4375Y-1.3974
N134X-.4428Y-1.3964
```

(Note: The NC file is truncated to avoid taking too many pages of this thesis)

VITA AUCTORIS

NAME:	Xinhui Zhong
PLACE OF BIRTH:	Guangzhou, China
YEAR OF BIRTH:	1968
EDUCATION:	South China University of Technology, Guangzhou, China
	1986-1990 B.E.Sc
	University of Windsor, Windsor, Canada
	2001-2003 M.A.Sc