

# **GLOBAL MANUFACTURING AND COLLABORATIVE TOOLS**

A Major-Qualifying Project Report

Submitted to the Faculty of

**WORCESTER POLYTECHNIC INSITUTE**

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

By:

Lincoln Bates      Caleb Oswald

In partnership with Huazhong University of Science & Technology students:

Tong Liuzhu    He Dan    Yang Xin    Zhang Chunjiang

August 16, 2010

APPROVED:

---

Prof. Yiming Rong, Ph.D.  
Project Advisor

# **ABSTRACT**

The project analyzes the current practices of Stanlok/REM and suggests means by which this corporation can increase global collaboration through the application of web-based engineering tools. Research was conducted in the field of software design, focusing on the communication between Java and SQL, along with preliminary study of relational databases. Proposed software was designed and implemented creating a server-based software tool which allows employees from any location to access and manage a manufacturing cost estimation dataset and deliver real-time simultaneous cost calculation.

# TABLE OF CONTENTS

Signature Page.....	1
Abstract.....	2
Table of Contents.....	3
Introduction.....	4
Background.....	7
About Stanlok/REM.....	7
Our Tasks.....	8
Engineering & Engineering Skills.....	9
Globalization.....	9
Typical Global Corporation.....	10
Global Engineering.....	11
Current Stanlok/REM Practices & Needs.....	12
Available Tools.....	13
Possible Solutions.....	16
Methodology.....	22
Results.....	23
Platform selection.....	23
Software design requirements.....	24
Software design process.....	26
Implementation.....	36
Conclusion.....	43
Bibliography.....	46

# INTRODUCTION

The HUST/WPI exchange program was established in 2004. This student exchange allows one of the finest engineering schools in both nations to collaborate on projects together. The project gives students the opportunity to broaden their global understanding. By engaging in the design process with their cross cultural partners, students are able to better understand how design and manufacturing is applied to a global economy. This global understanding is implemented by students working within the industry itself. They may be designing products, as they did in the 2005 program, or focusing more on the manufacturing side of the process, as students did in 2008, working with CIS. The benefits of this education are growing by the day as we approach an increasingly global world.

This project, Global Manufacturing Engineering & Collaborative Tools, is sponsored by Stanlok/REM. The focus of our study is therefore on understanding the current practices of its business and creating a way to improve the collaborative performance among four branches of the company. The general areas of study cover the following subjects: the background research of Stanlok/REM within the global marketplace, the understanding and analysis of its cost estimation Excel spreadsheet, and the research and implementation of supporting software - specifically, relational database management systems and the Java programming language.

The specific problems we are working to solve relates directly to the problems that any business, including Stanlok/REM, may potentially encounter. Currently, all of the manufacturing and material related data of the company are stored within an Excel spreadsheet on individual PC's, which leads to possible evolution of cost related data and equations, and data erosion over time. Another problem is information sharing. The cost model data is not fully available to all

manufacturing engineers and management within the global corporation, resulting in the different branches of Stanlok/REM working independently of one another, not collaboratively.

Our objectives to achieve improved global collaboration are as follows:

- 1) Understand the principles of engineering and the global engineering tools that are available, propose how these tools apply to the engineering process.
- 2) Understand the mode of current Stanlok/REM project work and determine what is needed to foster a more efficient global interaction.
- 3) Analyze cost estimation Excel spreadsheet provided by the company, including understanding the physical role of data and relational data equations.
- 4) Program data into a DataBase Management System, including researching DBMS software criteria and inputting data into SQL Server.
- 5) Build web based interface using Java, including creating Java PC interface and connecting interface to web SQL Server

As students majoring in mechanical and industrial engineering, we found the project difficult to understand at first because we knew little about business ideas, cost calculations, or computer database systems. Our biggest challenge was to learn the knowledge related to our objectives and make a clear task list, with thorough work distribution, so that we could *all* understand the concepts involved in completing the project. We searched for information in the library and on the internet and carried out a clearly defined work plan, with a detailed time table, to make sure our objectives were feasible and achievable in the short 8 week project deadline.

Ultimately, the result of this project is a software application to simultaneously calculate the delivered cost of a nut at four branches of Stanlok around the globe. We used Microsoft SQL Server to store and manipulate data, designed user interfaces and cost calculator functions with the Java programming language, and placed it on the internet so that the data would be uniform and shared amongst all of the authorized Stanlok and REM Wuxi Ltd. manufacturing engineers and management.

This solution will provide the framework to prove that increased information sharing is directly linked to improved collaboration between branches of a company, a result that can be applied to any business or engineering situation. Improved collaboration leads to more informed business and management decisions, and most importantly, breaks the current business trend of establishing “replica” branches around the world, all of whom compete against each other for profitable projects. Our solution will help to establish a foundation for a truly “globalized” and collaborative business structure through uniform cost estimation and real-time sharing of information.

# BACKGROUND

## ABOUT STANLOK/REM

Stanlok/REM Ltd. is a business that began over 80 years ago as the Worcester Taper Pin/Wright Machine Company (WTP), founded in 1929 by Frederick J. Barry. The Standard Lock Washer & Manufacturing Company (Stanlok) was founded in 1938 by a group of former WTP employees and competed with WTP until 1984 when Frederick Barry's son Al purchased Stanlok. Al Barry is a Worcester Polytechnic Institute alumnus, class of 1977. Stanlok and WTP existed in parallel until 1997, when WTP was forced to close its doors. Since the demise of WTP, Stanlok has absorbed WTP machines and employees and is now a leading manufacturer in standard and special taper pins. Wuxi REM Ltd. is a mechanical design, machining and forming company for contract customers. It is also owned by Al Barry, and as the name states, is located in Wuxi, China. They also design and produce nuts, bolts, pins and fasteners. Resources and knowledge are shared between the two companies. The Stanlok Group also includes New England Specialty Nut, Nuts Incorporated and Ramar Nuts Inc. The group is a master distributor of flat washers, tooth lock washers, split lock washers, helical spring lock washers, woodruff keys, dowel pins, cotter pins, spring/split pins, and many other fasteners in addition to taper pins.

Now you might be wondering, what exactly is a taper pin? A taper pin is a specialty fastener used in engineering applications. It is a steel (or whatever material being used) rod where one end has a larger diameter than the other. For example, a standard inch-sized taper pin has a taper of 1:48. This means that over the length of a 4 foot-long bar (48inches), the diameter of one end of the bar will be 1inch smaller than the opposite end (a 1/4inch taper over 1 foot of bar). Stanlok/REM designs and manufactures custom taper pins in odd lengths or diameters for

specialty engineering applications. This may also include threaded or notched pins to accept nuts, washers and many other types of industrial fasteners.

## **OUR TASKS**

As our goal states, we are seeking to help Stanlok/REM to become a more globally efficient organization. In order to do this we have come up with a few items which we feel must be addressed for our goal to be met. First, we want to create a system for Stanlok/REM that utilizes local talent while allowing projects to be projected across many branches of the company. By this we mean that every location operated by Stanlok/REM has a process or step of the process in which they specialize, but all locations are capable of undertaking entire projects if necessary. For example, a branch in Mexico might be better equipped for large volume production than the branch in Worcester, which specializes in design and accounting. Both branches are fully capable of designing and manufacturing any type of fastener required by the customer on site, but having the flexibility to perform any function in any locations allows for a more efficiently run company.

Secondly, we are seeking to shake the norm with our previous goal by creating a truly “globally collaborative” organization. Instead of creating replicas of the company at each location, we are trying to create a cohesive team setting where each team member is in a different location throughout the world. Improving collaboration and communication between branches are high priorities because they both help to foster teamwork and make project sharing simpler. The idea is to improve upon existing processes to increase manufacturing options and increase overall performance of the company.



## **ENGINEERING & ENGINEERING SKILLS**

Engineers and individuals in technical fields must be able to see the big picture. What is the problem? What is required to fix the problem? And how to go about implementing solutions? In order to understand the process of engineering and problem solving, individuals must have a specialized knowledge of the problem or process they are analyzing. For example, placing a layperson in a position to make decisions regarding the design of a nuclear power plant would end with nothing being achieved because that person would not even know where to begin. Engineers must be able to analyze a problem and decompose it into manageable pieces. They also must be able to understand the math and science behind every one of those pieces in order to improve and optimize the individual sections and the problem as a whole. The engineer must be able to interpret data, and understand cost, quality, safety, ethics and how their work fits into society and a business environment. Most importantly, especially now in the growing global and collaborative engineering markets, an engineer must be a “people person”. He/she must be able to manage AND work in an interdisciplinary team effectively, communicate objectives and ideas clearly, and while being part of a team, maintain individual creativity, innovation and self-motivation.

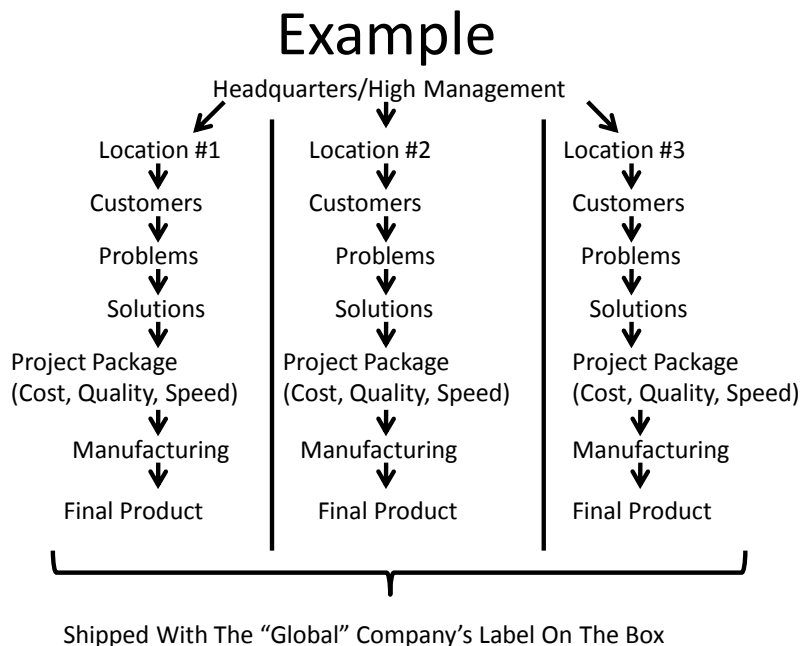
## **GLOBALIZATION**

Globalization is the process of connecting the international marketplace. The goals of globalization are to enable a more efficient exchange of goods and services in a cross cultural system. By providing means of communication and a networking of knowledge businesses are better able to interact and meet the needs of their clientele. This process has been successfully realized in several businesses. The work of IBM is a key example of this. IBM provides

services that allow businesses to be fully connected across their various branches by establishing networks of business knowledge. While IBM designs systems of information management, the tools necessary to create these systems are provided by software companies, such as Microsoft.

## TYPICAL GLOBAL CORPORATION

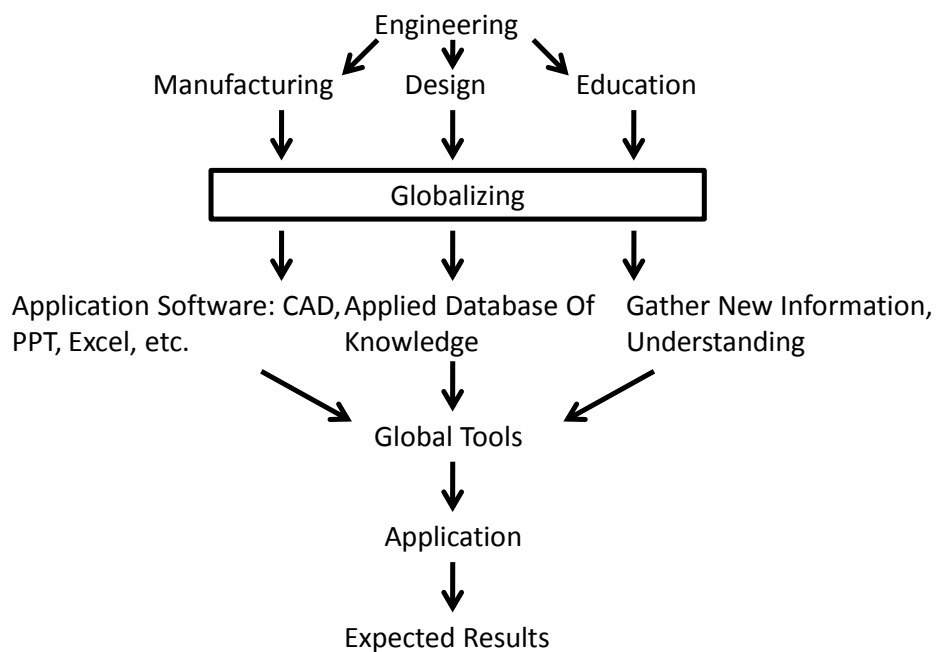
Many businesses claim that they are global. In reality, these corporations often have mere clones of their original business model distributed internationally. The business planning is overly centralized and does not incorporate the variables affecting each of their branches effectively. Separate branches will attempt to meet the same need of their client, will come up with separate and incompatible designs, which fails to utilize the benefits of a global business community to their fullest extent.



*Fig. 1: Typical Global Corporation Flowchart*

## GLOBAL ENGINEERING

What this project aims to do is to apply the same framework of the IBM Microsoft business model to the design process. Stanlok/REM is an international design and manufacturing firm that operates within several different markets and owns facilities better equipped to perform different aspects of the engineering process. By compiling a database of engineering knowledge, providing the tools necessary to manipulate and work with this knowledge, and suggesting optimal means of communication and education, this project hopes to allow this corporation to better function in a global market. This incorporation of engineering knowledge in the global business model will, hopefully, allow for the creation of an effective global engineering process.



*Fig. 2: Ideal Global Corporation Flowchart*

## **CURRENT STANLOK/REM PRACTICES & NEEDS**

Primary to the background research of this project is an analysis of Stanlok/REM's current business practices, and the areas which Al Barry feels the company needs improvement. Currently, manufacturing is a single location procedure. A factory receives an order, based primarily on the amount of product to be produced, and manufactures the parts fully within their local markets. In order to maintain an awareness of what is occurring within his business, Al Barry is forced to travel to these locations in order to understand what is occurring within those locally centralized facilities. Information regarding projects and processes is relayed through individual emails. The potential for limitations regarding available machinery, higher costs due to local market forces, and inadequate communication of information is high.

The needs which must be addressed in order to alleviate these issues are as follows. Information regarding the specifications of pending and past projects must be made available on a worldwide scale. This information will allow all locations to better understand the projects which they are manufacturing. Factors such as the types of machining necessary, the time required for machining, and the total costs of production should all be compiled along with any other pertinent information. A system which quantifies the production possibilities of each factory should also be created. The various strengths and weaknesses of each branch should be readily available in order to understand the optimal use of the facilities available to the business. This information should be available to be readily analyzed in order to optimize the efficiency of delivering to the customers the highest quality product, in the shortest time and at the lowest costs. In order to facilitate this compiling of information, communication beyond emails must be established. All locations should be able to freely share and discuss all information regarding the specifics of any product which is being produced, ultimately, by the company as a whole.

## **AVAILABLE TOOLS**

In order for us to come up with a comprehensive plan to increase Stanlok/REM's efficiency and collaboration, we must first understand the tools available for us to use. To increase communication and collaboration between PC-based individuals, the internet and a few web-related communication tools can be utilized.

The first web-based tool available to us is an online database of knowledge. Having the ability to see project progress and ideas in real-time allows for faster communication of new solutions and entire projects. Instead of having to wait for a colleague to email you a part of the project for you to continue, it would be ideal for updates to be available in an online workplace. This would also help management track project progress in order to be more timely and accurate in communicating project completion and any problems to customers.

Some internet tools to foster better communication include remote desktop connectivity, over-the-internet telephone, video and multiple location conference. However, it is peer-to-peer networking, most specifically file hosting services, that make the most sense in the context of our project goals. These types of services allow multiple users to upload files to an internet accessible server for viewing and editing by anyone with the log-in credentials to access the database. File hosting services allow for private interactions within a web-based system for improved business databases. A popular example of a private user file hosting service (not meant for business, but simply single users) is GoogleDocs. GoogleDocs is a free service that allows users to upload documents, spreadsheets, presentations, drawings and forms to a password protected location to be viewed from any internet connection or "smart" cellular phone. The homepage of GoogleDocs even states "Share Your Work: Real-time collaboration

means work gets done more quickly.” Sounds exactly like the kind of file sharing database that we need to meet Stanlok/REM’s needs.

The web-based database should also include up-to-date market material and machining costs for each location so that a manager could determine the best location in which to send the project. Determining costs is very complicated. One must create a summation of material, labor, manufacturing, and shipping costs in order to determine the total cost of creating a part for a customer. Material costs are easily found through a distributor’s public information or by picking up the phone to obtain a direct quote. Shipping costs are also simple to determine, as one can also obtain a quote from any number of shipping companies and organizations. Labor costs are more complicated. They vary depending on the amount of time it takes to create parts, as hourly wages depend on the amount of time required. Similarly, manufacturing costs are determined by the complexity of the part, the quality required (it costs more to machine a part to .0001” tolerances than to machine to .01” tolerances), and once again, the amount of time required to machine the parts. There are a number of equations available publicly to determine individual and total business costs. Al Barry also has his own proprietary Windows Excel-based cost calculation spreadsheet that may fit the bill if modified for a company-wide standardized cost comparison system in the web-based database. This spreadsheet will be discussed in detail a little later.

Another comparison tool is time evaluation. Customers want a quality part for as little money as possible as quickly as possible. In order to determine the total expected time required to manufacture a single part and a whole order of parts, more equations are utilized. They involve the time required to make certain cuts and to perform specified manufacturing processes, in addition to tool change time (required when either a tool dulls or a different kind of cut is

required). The total tooling and manufacturing time involves many variables and will obviously be different for different Stanlok/REM locations due to machines of varying speed and complexity. Being able to compare tooling and manufacturing time between individual branches through the internet database allows engineers to determine the best location in which to balance the three points of the Project Triangle, optimizing for cost, speed and quality.

The last point of manufacturing, quality, requires more in depth analysis. Determining quality quantitatively for comparison requires the use of several quality control measures. Understanding human, machine and process limitations allows us to empirically calculate the manufacturing performance at each branch of Stanlok/REM. For instance, machines that manufacture for tighter tolerances will allow customers who require those tolerances to buy from Stanlok/REM. Or if the parts are required for a very short deadline, perhaps the location with the most machines may be the best choice. Analyzing performance objectively allows once again for optimization and selection of branches to complete the work that qualifies to meet the needs and demands of the customer. Due to the integrated nature of “quality” in part manufacturing specification and sizing, particularly tool selection, we will assume from this point on that objective quality quantification is encompassed within manufacturing.

And that’s how a customer-based business is run. First, the need of the customer or problem that they have must be understood. Next, the company must create a work package or estimate of the total cost and the time required to perform the work, hopefully to as high a quality as possible. Lastly, the customer will accept, reject or modify the package to best fit their needs. This is the most important part of not just customer service-based engineering but all engineering in general: does the solution fit the need? Without proper analysis and understanding of the steps required to make that determination, no work can be done. Customers

require that their needs will be met, or they will go find someone who can meet those needs with the highest quality parts, on a feasible budget, in the shortest amount of time.

## **POSSIBLE SOLUTIONS**

We have researched a few of the most popular data management software and programming solutions. After learning about some of the possible ways to solve our problem, we will discuss which platform(s) we chose and why we chose them, using criteria supplied to us by Stanlok/REM and its business strategies.

### *Microsoft Access*

Microsoft Access is a relational database system embedded into the Microsoft Office suite of applications. Access is considered a “pseudo-relational database management system” as it contains a simple database engine (called Microsoft Jet) and a graphical user interface, but lacks the ability to independently extend outside the Microsoft Office family. Data can be imported directly from applications such as Excel, Outlook or SharePoint, just to name a few.

The limitation to Microsoft Office also allows MS-Access to be an extremely powerful tool when used within the Office family. Using the Office-wide Visual Basic programming language, connectivity and simplicity between Microsoft software allows beginning users to create a simple and effective database management system.

However, this ease of use is a hindrance when advanced users need a more powerful database tool. The Visual Basic language and Access tools don't allow enough freedom to build applications outside of the Office software suite (such as when creating web applications). For



personal or very limited business use, Microsoft Access is a simple database construction and manipulation solution.

### *Microsoft SharePoint*

Microsoft SharePoint is collaboration and web publishing software that can be used to create web sites, search engines, intranets and more. SharePoint is not intended to replace a full scale file server system, but can be implemented in small businesses or as internal business applications. SharePoint is a part of the Microsoft Office suite and is built to work in conjunction with those programs.

SharePoint is a simple web publishing application and is intended to be implemented without significant IT requirements. Again, like MS-Access, being simple and user friendly, in combination with Office compatibility is both a blessing and a curse. Beginners can create business solutions rapidly, but advanced developers may find SharePoint frustratingly basic. For instance, while SharePoint can be used with any internet browser, only Internet Explorer (a Microsoft product) allows the use of all of the SharePoint functions. Mac users utilizing the Safari web browser or Windows users currently using Firefox, will not be able to implement SharePoint to its fullest potential. This limits the functionality of an internet program that can be accessed by any user using any type of internet browser.

### *SQL Server*

SQL Server 2005 is a comprehensive database platform which uses the tool of integrated business intelligence (BI) to provide large-scale business data management. It was jointly developed by Microsoft, Sybase and Ashton-Tate originally and the first version was

introduced in 1988. After the introduction of Windows NT, Microsoft and Sybase have been developing SQL Server in separate ways. Microsoft ported SQL Server into Windows NT, and focused on promoting the Windows NT version of SQL Server while Sybase focused on the application of SQL Server in the UNIX operating system.

SQL Server 2005 is a Microsoft product. Its database engine is the core of the database solution. It provides secure storage for relational data and structured data, making it possible for its users to construct and manage high-availability and high-performance data applications for business. In addition, SQL Server 2005 is also integrated with functions for analysis, reporting, integration and notification. It enables the user's company to construct and deploy cost-effective solutions and apply the data applications to every field of the business through servers, web services and mobile devices.

## *Oracle*

The title "Oracle" applies to both the company and its proprietary database software. Oracle Corporation is one of the world's leading database development companies, with a number of software and hardware products for database construction, maintenance, and use. Their products include Oracle Database software, Oracle Solaris server and storage hardware systems, and the Java programming language (after the acquisition of Sun Microsystems in January 2010). The latter will be the focus of further explanation later on, but for now we will discuss the Oracle Database (also called Oracle RDBMS or simply Oracle).

Oracle software was created in 1979 by Larry Ellis, Bob Miner and Ed Oates. The name came from a CIA database management system project that the Ellis had been working on. The project was quickly scrapped, but the men saw business potential in the new computer system

architecture and started Software Development Laboratories (SDL) in 1977. SDL later became present day Oracle Corporation. Oracle was the first RDBMS software that was commercially available to the public, and also the first available to the public using the SQL programming language developed at IBM by Donald Chamberlin and Raymond Boyce in the early 1970's. IBM had been using the SQL language for military and government applications. Even today, IBM and Oracle battle for the top spot in business database management.

Oracle requires extensive IT support and is not open source, meaning that one must pay for a license agreement and continues to pay fees over time in order to utilize its services. While Oracle may be a very robust platform, it is complicated and more difficult to use than other options. It is capable of tackling any database management a business might have, but the limited access and amount of money required to create and maintain an Oracle system makes it a less than ideal option for many small businesses and for personal use.

### *Java*

Java is a programming language originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are compiled in code that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is general-purpose, concurrent, class-based, and object-oriented, and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere". Java is currently one of the most popular

programming languages in use, and is widely used from application software to web applications.

One characteristic of Java is portability, which means that computer programs written in the Java language will run similarly on any supported hardware/operating-system platform. This is achieved by compiling the Java language code to an intermediate representation called Java byte code, instead of directly to platform-specific machine code. Java byte code instructions are different than machine code, but are intended to be interpreted by a virtual machine (VM) written specifically for the host hardware. End-users commonly use a Java Runtime Environment (JRE) installed on their own machine for standalone Java applications, or in a Web browser for Java applets (small applications built for web use).

### *Eclipse*

Eclipse is an IDE (integrated development environment) originally developed by IBM to replace its commercial software called Visual Age for Java. It is an open source product and Java-based development platform. In essence, Eclipse is merely a framework and a set of services that create a development environment for constructing plug-ins. Eclipse comes with a standard plug-in package, including the Java Development Tools, JDT.

Eclipse focuses on providing a full-featured and commercial quality platform for highly integrated Java development. It comprises 3 projects - Eclipse Project, Eclipse Tools Project and Eclipse Technology Project and includes 4 parts - Eclipse Platform, JDT, CDT and PDE. JDT supports Java development, CDT supports C development, and PDE supports plug-in development. Since the Eclipse platform is an open source IDE, it provides a public development

platform, with functions for constructing blocks and running integrated software development tools. Eclipse Platform allows programmers to develop independent tools which could be seamlessly integrated to other ones without distinguishing where the function of a tool ends or where another tool begins to work.

# METHODOLOGY

The specific nature of the project is not to directly solve a problem at Stanlok/REM. Instead we will try to create a solution for a future problem that could potentially occur at any organization, including Stanlok or REM. Many times, when cost estimation is performed, even a standardized, company-wide cost estimation template will evolve over time depending on who is performing the estimation. There will always be some inherent degree of subjectivity in the decisions that affect the final delivered costs. That is to say that if two engineers sitting next to one another perform a cost estimation for the same part order, the final calculated costs will most likely be slightly different, depending on the decisions each engineer made, according to his or her level of experience or knowledge of the manufacturing process and the current market.

We plan to create a standardized cost estimation application that eliminates as many subjective decisions as possible by building the estimation program “in the cloud” and making it a web-based program. We plan to incorporate a database of related costs (materials, machine rates, etc.) that can only be manipulated by management in the hopes that the variability in those costs over the course of time will not be a significant factor in the final costs. Any manufacturing engineer or management employee will be able to access the program from any location with an internet connection and enter the customer supplied sizing, material, and required documentation information and the final delivered cost will be uniform no matter the user’s knowledge level. We will prove this concept by building an application to calculate the cost to manufacture a nut, of any typical type, size, or material.

# RESULTS

## PLATFORM SELECTION

Before we delve into the complex details of manufacturing cost estimation, we must first decide which software platform(s) we will be projecting our program on to. We broke the decision process into three distinct criteria, cost, complexity and clarity, with which we hope to optimize in our solution.

The most critical software selection criterion is cost. The software should be inexpensive and avoid any unnecessary licensing or renewal fees. In addition, the platform must be simple enough that extensive IT support is not required, an additional cost that we are trying to avoid. As a consequence of limiting IT support and its related costs, management (or any user) must be able to perform the necessary operations and manipulate the related data without prior, or advanced, programming or computer knowledge. The software must be robust enough to perform the cost estimation efficiently, while remaining user friendly. Finally, the inputs and related outputs must be clear to any user of any knowledge level, so that accurate information may be obtained through the use of the software.

Using this selection criterion, we chose to use Microsoft SQL Server 2005 as our database background and a Java PC interface in the foreground, connected by Java's integrated DataBase Connectivity (JDBC). Microsoft Access, with its Jet database engine, is not powerful enough to manage our data, especially in a web environment, where Access simply cannot be used to its full potential unless the user has Microsoft Internet Explorer, and Oracle is too complicated and expensive for our purposes, requiring advanced programming knowledge (IT support) and a costly license agreement.

Settling in the middle of the two, SQL Server fits the bill for our database management system. It is an open source product, meaning that it's free to download and use, and has been in the public domain for some time now, where it has proved that it can be effectively used in conjunction with the Java language and interface. The Java programming language is also an open source product and is powerful enough to build the proper user interface and data foreground to meet our needs.

## **SOFTWARE DESIGN REQUIREMENTS**

Now that we have selected the software foundation for our program, we must begin to design the framework for it to sit within. Designing a building piece by piece before even breaking ground is the best way to ensure that the final product will be as strong and effective as initially penned. In our case, the final product must contain the following pieces: a Java graphical PC interface, a web-based SQL Server database, and a JDBC bridge between the two.

The PC interface is the most visible part of our product. Every user will interact with the PC interface every time the program is opened. It must be easily understood, meaning that its layout is critical to an enjoyable and effective cost estimation experience. It should include drop-down menus and customer specification inputs, the user should be able to manipulate (add, modify or delete) parameters in the estimation calculator, and most importantly, the PC interface must be able to produce reliable cost estimations, not just of total cost, but also of individual manufacturing, materials and machine costs. This information will not only be essential to the customers decision-making process, but to also aid the bidding company to evaluate its efficiencies.



Next, the web-based server database must be able to store and manage all data necessary to perform the cost calculation. This includes material costs and properties, machine rates and specifications, and tool selection and sizing, to name a few. In the case of Stanlok/REM, the company has an Excel cost estimation spreadsheet that is currently in use, which contains the necessary data fields and specific data that are required for cost estimation. The details of the Stanlok/REM Excel estimator will be explained later on in this report, but for now we will focus on the general idea of what our database will need.

The data within the database must be manipulatable. For instance, as market forces change the price of materials, that change must also occur in our database as well. This means that management, or whomever has the authority to do so, must have the ability to change the data values from anywhere in the world. Creating a web-based server database allows the data to be changed online, and those changes will be seen from all of the connected PC interfaces in real time. The add, modify and delete functions performed on the Java interface must manipulate the data in the SQL Server to create an accurate and up-to-date cost estimation program.

This will be the job of the JDBC. Connecting the actions that occur on the Java interface (inputs, changes, etc.) to the actual data in the server will create the physical calculations of the costs. The relational equations and their resulting calculations, between material properties and machinability for instance, will occur in the JDBC portion of the program. The JDBC's job is to connect the variables of the cost estimation equations *to* those equations in order to add functionality to our Java interface and produce a useful calculated output.

## **SOFTWARE DESIGN PROCESS**

### Cost Model Analysis

The first step in building a relational database system is to understand the data itself. At the outset of this project we received an Excel spreadsheet that was currently being used at Stanlok/REM's various factories. This spreadsheet is a cost analysis calculator for orders received by Stanlok/REM which included machining and material costs, as well as special treatments data and additional industry specifications. By analyzing this Excel spreadsheet, the use of the various data fields becomes clearer. An initial overview provides the user with an understanding of what data sets represent physically. Further analysis allows the user to understand the role of this data in cost estimation form. We will explore the calculating pages of the spreadsheet, break them down, and represent the cost estimation model in terms of general equations that apply to every cost model in an effort to explain the "behind the scenes" calculator functions that will be used in our new software. All of the equations expressed and explained have been taken directly from the functions of the Stanlok/REM Excel spreadsheet, and may or may not be common knowledge equations. They are simply the result of our analysis of Stanlok/REM's current cost model. The summary and calculation pages of that model can be seen below.

Quote No.	34-3116-1624-AA			Estimator	AA	8/3/2010
Customer				PPAP Level 3		No
RFQ No.				Cert for Material, Plating		Yes
Drawing No.				Proof Load, Hardness		No
Contact				Mag Particle, Mag Perm SS		No
Phone No.				Fluor. Liquid Penetrant		No
Fax No.			MM	Thread Size		#08
eMail			4.17	Thread Decimal		0.164
Material	Stainless, 304 (A2)		1.06	Thread Pitch		24
Plating?	NONE			Thread Form		Metric
Heat Treat?	NONE			Thread Class		6H
Units	Inch			Thread Hand		RH
Shape	Hex	MM		Thread, Re-Tap?		No
AF or Dia	0.313	7.94		Flats? No.?	No	None
Thickness	0.157	4.00		Slots? No.?	No	None
Nut Type, Spec.	OTHER			Holes? No.?	No	None
Note				H. Wire? No.?	No	None
Tool\$, Gage\$ ?	\$ 100.00			Stamp?	Yes	1
<b>Total Cost per 100</b>	<b>Cost Elements</b>	<b>Qty 1</b>	<b>Qty 2</b>	<b>Qty 3</b>	<b>Qty 4</b>	<b>Qty 5</b>
	Quantity	1,000	2,500	5,000	10,000	-
	Material Qty Lbs	5.8	14.5	29.1	58.1	-
	Material Cost (\$C,1 Bar, \$50)	5.00	2.03	2.03	2.03	50.00
	Manufacturing Cost (\$/C)	12.85	12.85	12.85	12.85	12.85
	Set Up, Lot Charges (\$/Lot)	17.50	7.00	3.50	1.75	175.00
	Plating, Finishing (\$/lot, \$/C)	-	-	-	-	-
	Heat Treating (\$/Lot, \$/C)	-	-	-	-	-
	Tooling, Gage Charges (\$/Lot)	10.00	4.00	2.00	1.00	100.00
	Total Cost \$/Lot or \$/100	45.35	25.88	20.38	17.63	-
	Gross Margin Target	40%	35%	30%	25%	20%
	Sales Price \$/100, Calculated	\$ 75.60	\$ 39.90	\$ 29.20	\$ 23.60	\$ -
<b>Sales Price</b>	<b>Sales Price \$/100, Actual</b>	<b>\$ 76.00</b>	<b>\$ 40.00</b>	<b>\$ 39.00</b>	<b>\$ 23.50</b>	<b>\$ -</b>
	Sales Price \$/Each, Actual	\$ 0.76	\$ 0.40	\$ 0.39	\$ 0.24	\$ -
	Order Value	\$ 760	\$ 1,000	\$ 1,950	\$ 2,350	\$ -

Fig. 3: Summary Page

Manufacturing Cost Estimate						
Quote No.	34-3116-1624-AA					
Material and Speed Estimate	Description	Calculation		Desc.	Calc.	
	AC	0.361		Mtl \$/Lb	\$ 3.50	
	Machine	#00 B&S		Lb/piece	0.006	
	Cut Off	0.063		Lb/100	0.58	
	SFPM	75		Pcs/Bar	620	
	RPM	794		Lbs/Bar	3.60	
	Length	0.232		FG Lb/100	0.35	
Automatic Time Estimate	Operation	Feed Rate	Length	Revs	Sec	Total Sec
	Feed stock				3	3
	Drill, center	0.0020	0.060	30	2.3	5.3
	Drill, tap	0.0030	0.250	83	6.3	11.6
	Tap		0.250		0.0	11.6
	Form chamfer (overlap)	0.0015	0.040	27	2.0	11.6
	Cut off	0.0020	0.091	45	3.4	15
	<b>Total Time, seconds</b>				185	18
Manufacture Cost Estimate	Department	\$Mach/Hr	Prod/Hr	Cost/100		Set-Lot \$
	Blank, automatic (75%) (\$/C)	\$ 10.00	180	5.56		\$ 50
	Countersink (\$/C)	\$ 20.00	800	2.50		\$ -
	Deburr, (\$/Lot)					\$ 50
	Tap (\$/C)	\$ 20.00	640	3.13		\$ -
	Flats, mill (\$/C)	\$ 20.00	0	-		\$ -
	Slots, mill (\$/C)	\$ 20.00	0	-		\$ -
	Drill Holes (\$/C)	\$ 20.00	0	-		\$ -
	Mark or stamp (\$/C)	\$ 10.00	600	1.67		\$ 50
	Re-Tap (\$/C)	\$ 10.00	0	-		\$ -
	Wire holes (\$/C)	\$ -	0	\$ -		\$ -
	Documents (\$/Lot)	Yes				\$ 25
	<b>Total Mfg Cost/100</b>			<b>\$ 12.85</b>		<b>\$ 175</b>

Fig. 4: Calculation Page

## Material Equations

Calculating material costs is the simplest of our estimation equations. The following general equation governs material cost in the Excel document:

$$M = \left(\frac{q}{p}\right)(ud)$$

Where:  $M$  = Total material cost

$q$  = quantity of parts requested

$p$  = number of pieces per bar

$u$  = lbs per bar

$d$  = U.S. dollars per lb

This general equation looks simple at first glance, but break it down piece by piece and it gets slightly more complex. The number of parts requested ( $q$ ) is a given value, received from the customer. The number of pieces per bar ( $p$ ) is determined by the size and shape of the part, and also the size of the bar of stock material. It can be expressed as:

$$p = \frac{V}{v}$$

Where:  $V$  = volume of bar

$v$  = part volume

If the size of the part increases and/or the size of the stock material decreases, then  $p$  will decrease and the final cost will go up (more bars means more lbs of material, which costs more money). Volume can be estimated geometrically and as a summation of multiple smaller volumes. It can be expressed as:

$$V = \text{volume of bar} = \text{volume of a cylinder} = \pi r^2 l$$

$$v = \text{part volume} = \sum(v)_1 + (v)_2 + (v)_3 + \cdots + (v)_n$$

The weight of each bar ( $u$ ) is determined by the volume of the bar of stock material multiplied by the density. Using the bar volume equation previously discussed and multiplying by the density, we will have the weight (in lbs) of each bar. This is expressed as:

$$u = V \cdot \rho$$

Where:  $V$  = volume of bar =  $\pi r^2 l$

$\rho$  = material density

Finally, the material cost per lb ( $d$ ) is a fixed value, which does change over time, but is considered a constant in this equation. With all of these pieces we can build an equation that better explains the cost of materials in our cost model. It looks like this:

$$M = \left( \frac{q(\sum v)}{\pi r^2 l} \right) (V \cdot \rho) d$$

## Manufacturing Equations

Calculating manufacturing costs are extremely complex. Again, we will begin by discussing a general equation and breaking it down into parts before building a more specific mathematical model. The total manufacturing cost can be described as a summation of time of each operation multiplied by the machine rate (dollars per hour) to perform the operation. The general equation is as follows:

$$F = \sum(t_1 \cdot C_1) + (t_2 \cdot C_2) + (t_3 \cdot C_3) + \dots + (t_n \cdot C_n)$$

Where:  $F$  = total manufacturing cost

$t$  = total time required to manufacture parts

$C$  = machine rates

Machine rates are proprietary to every machine and every factory, meaning that the calculation of machine rates is entirely a business analysis, of costs per hour to operate a machine, which includes utilities, labor, initial machine purchase price, and any monetary variable that affects the

purchase and use of a machine. We won't get into this determination, as every business may have a different way of calculating its own machine rates.

We will however explore how manufacturing time is calculated. Part size and material selection are the major variables in determining manufacturing time. To begin, the across-the-corners size (we will call it  $S_{AC}$ ) of the requested part determines the machine and machine tool, which in turn determines the cut off length, total length, machine surface feet per minute (SFPM), tool RPM and feed rates. Depending on the size of the part being manufactured, these variables will change accordingly. Now that we have introduced some machine parameters, we can begin analyzing the individual operations.

The complexity of the part will determine the operations required to manufacture the part. Every new order will require a different set of operations to make the parts and fill the order. However, our cost model will only be determining the cost to manufacture a nut, of any typical size, shape and material. Therefore, we will focus on the operations of feeding, center drilling, tap drilling, form chamfering and cut off, as these are the operations required to make a nut.

The feeding operation time (we'll call it  $t_f$  for convenience) is a set time value unique to a machine and will be considered a time constant from here on. Determining center drilling time uses the following equation:

$$t_{cd} = \frac{R}{\left(\frac{RPM}{60}\right)}$$

Where:  $t_{cd}$  = center drilling time (secs)

$RPM$  = machine revolutions per minute (unique to each machine)

$R$  = part revolutions

Part revolutions are determined by dividing the length of the cut ( $l$ ) by the drill feed rate ( $f_d$ , in seconds). Drill feed rate is determined by material selection and machine properties, so it will be

considered a given constant for this equation, but we must remember that it varies between materials and machines. The length of the cut, however, is determined in the following manner:

$$l = .005 + \left(\frac{d_c}{2}\right) - \frac{\left(\frac{d_t}{2}\right)}{1.664}$$

Where:  $l$  = length of cut

$d_c$  = countersink diameter

$d_t$  = tap drill diameter

The countersink and tap drill diameters are determined by the thread decimal selection, a given size constant, decided by the customer specifications of the part size. Combining all of these expanded terms, we get:

$$t_{cd} = \frac{f_d \left(\frac{RPM}{60}\right)}{\left(.005 + \left(\frac{d_c}{2}\right) - \frac{\left(\frac{d_t}{2}\right)}{1.664}\right)}$$

Next we will investigate tap drilling. Again, feed rate is unique to the machine and the tap drilling operation, so we will call it a constant,  $f_t$ . Similar to center drilling, the following equation governs the tap drilling operation:

$$t_{td} = \frac{R}{\left(\frac{RPM}{60}\right)}$$

Where:  $t_{td}$  = tap drilling time (secs)

$RPM$  = machine revolutions per minute (unique to each machine)

$R$  = part revolutions

Also, again part revolutions are equal to the length of the cut divided by the tap drill feed rate ( $f_t$ ). In tap drilling however, the length of cut is expressed as:

$$l = 0.01 + l_T$$

Where:  $l_T$  = total length

Total length is dependent upon customer specifications and machine tool selection. It is expressed as:

$$l_T = (W_p + W_{co}) \times 1.03$$

Where:  $W_p$  = part width

$W_{co}$  = cut off width

Combining all of the expanded terms we get:

$$t_{td} = \frac{f_t \left( \frac{RPM}{60} \right)}{0.01 + 1.03(W_p + W_{co})}$$

The next operation is form chamfering; here once again the time to perform the operation is expressed as:

$$t_{fc} = \frac{R}{\left( \frac{RPM}{60} \right)}$$

Revolutions are calculated as by the length of the cut divided by the chamfering feed rate ( $f_c$ ).

The length of cut is expressed as:

$$l = 0.01 + \frac{S_{AC} - d_p}{2}$$

Where:  $S_{AC}$  = across the corners size

$d_p$  = part diameter

Combining all of the investigated terms, we get:

$$t_{fc} = \frac{f_c \left( \frac{RPM}{60} \right)}{\frac{S_{AC} - d_p}{2}}$$

Lastly, the cut off operation follows along the same family of equations, where:

$$t_{co} = \frac{R}{\left( \frac{RPM}{60} \right)}$$



And

$$l = \frac{d_P}{2} - \frac{d_t}{2}$$

Where:  $d_t$  = tap drill diameter (introduced in the center drilling equation above)

$d_P$  = part diameter

Combining all of the terms, we get:

$$t_{co} = \frac{f_{co} \left( \frac{RPM}{60} \right)}{\frac{d_P}{2} - \frac{d_t}{2}}$$

Combining all of the general equations, total manufacturing time can be shown as:

$$t = t_{cd} + t_{td} + t_{fc} + t_{co}$$

Or, expanded as:

$$t = \frac{f_d \left( \frac{RPM}{60} \right)}{\left( .005 + \left( \frac{d_c}{2} \right) - \frac{\left( \frac{d_t}{2} \right)}{1.664} \right)} + \frac{f_t \left( \frac{RPM}{60} \right)}{0.01 + 1.03(W_p + W_{co})} + \frac{f_c \left( \frac{RPM}{60} \right)}{\frac{S_{AC} - d_P}{2}} + \frac{f_{co} \left( \frac{RPM}{60} \right)}{\frac{d_P}{2} - \frac{d_t}{2}}$$

Thus, the total manufacturing cost for a nut using our model is expressed as:

$$F = \left[ \frac{f_d \left( \frac{RPM}{60} \right)}{\left( .005 + \left( \frac{d_c}{2} \right) - \frac{\left( \frac{d_t}{2} \right)}{1.664} \right)} \times C_{cd} \right] + \left[ \frac{f_t \left( \frac{RPM}{60} \right)}{0.01 + 1.03(W_p + W_{co})} \times C_{td} \right] + \left[ \frac{f_c \left( \frac{RPM}{60} \right)}{\frac{S_{AC} - d_P}{2}} \times C_{fc} \right] + \left[ \frac{f_{co} \left( \frac{RPM}{60} \right)}{\frac{d_P}{2} - \frac{d_t}{2}} \times C_{co} \right]$$

## Labor & Shipping Equations

We cannot determine total delivered costs without discussing *all* aspects of cost, including labor costs and shipping costs. For our purposes, we will assume labor costs to be built into the machine rate, as it incorporates all aspects of running a machine per unit time. We will neglect shipping costs as well, because with many shipping companies now offering flat

rates on standard sized shipping boxes, shipping costs can quickly and easily be added to our final cost output for an accurate estimate of total delivered cost.

## Total Cost

The total delivered cost of a nut can be expressed generally as:

$$TOTAL = M + F + S$$

Where:  $M$  = total material cost

$F$  = total manufacturing cost

$S$  = total shipping cost

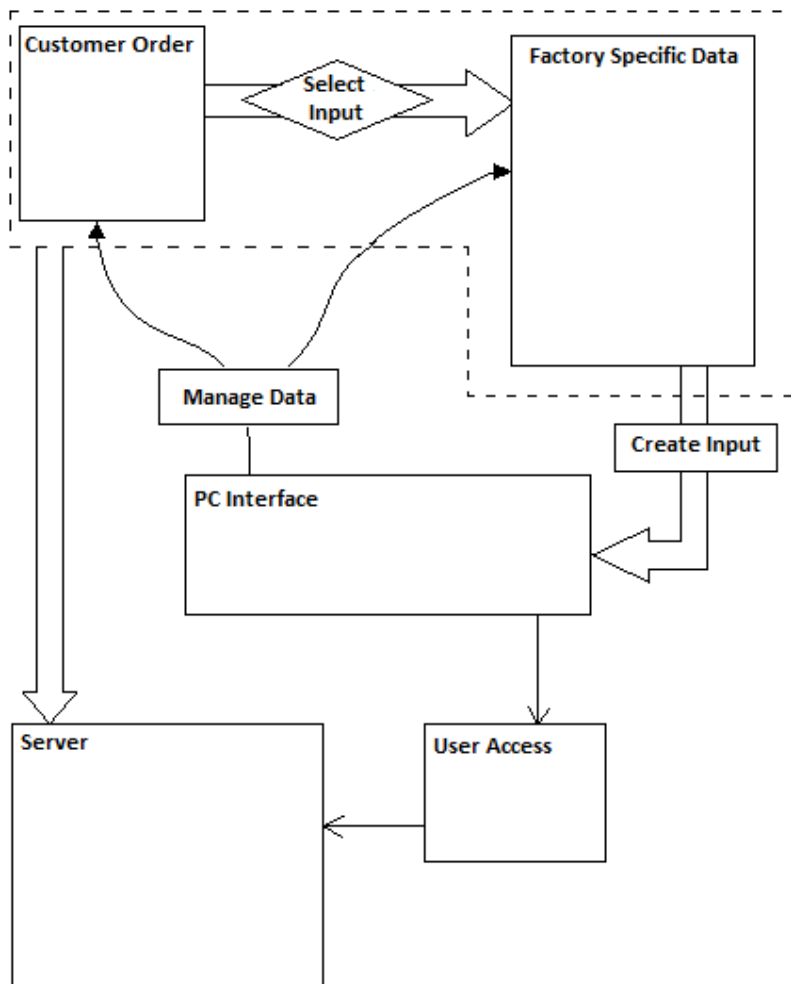
The fully developed model of total delivered cost, including all expanded equations examined above looks like this:

$$TOTAL = \left( \left( \frac{q \cdot (\sum v)}{\pi r^2 l} \right) (V \cdot \rho) d \right) + \left[ \frac{f_d \left( \frac{RPM}{60} \right)}{\left( .005 + \left( \frac{d_c}{2} \right) - \left( \frac{d_t}{2} \right) \right)} \times C_{cd} \right] + \left[ \frac{f_t \left( \frac{RPM}{60} \right)}{0.01 + 1.03(W_p + W_{co})} \right] + \left[ \frac{f_c \left( \frac{RPM}{60} \right)}{\frac{S_{AC} - d_p}{2}} \times C_{fc} \right] + \left[ \frac{f_{co} \left( \frac{RPM}{60} \right)}{\frac{d_p}{2} \cdot \frac{d_t}{2}} \times C_{co} \right] + S$$

We will be using these equations as relations in our Java PC interface to build the calculator functions. They will make up the “meat” of our program, but will not be physically seen by the user. The calculator functions built in Java will remain the same as the calculator functions used in the Excel spreadsheet.

## Software Model

The diagram below illustrates our software design plan. It describes a centralized Java PC interface that will allow the user to input the customer order, manage the factory specific data, and control user access. The customer order will determine what data will be used for the calculation, which will be input into the user interface and costs determined. All data will be placed on the internet on a custom SQL Server 2005 server for manipulation as needed. Any changes performed on the server will be seen from any PC interface.



# IMPLEMENTATION

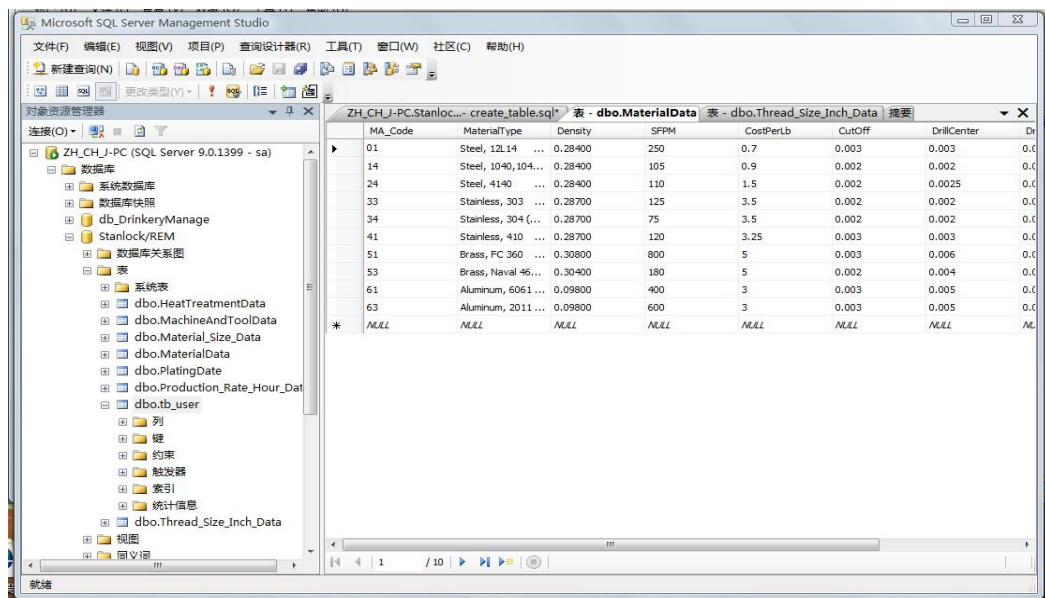
## Building the SQL Server

Having been given a cost estimation model already in use at Stanlok/REM, we started by compiling the data to be populated in the SQL server. The data fields used for the Excel spreadsheet calculation will also be used in our software, so we were able to transfer the data already in the spreadsheet to the SQL server. For example, we converted the following Excel data field into SQL code and the resulting table:

Fig. 5: Material Properties-Excel

Material Properties					Auto Lathe Feed Rates				
Material Type	Code	Density	SFPM	Cost/Lb	Cut Off	Drill Center	Drill Tap	Form	
Aluminum, 2011	63	0.098	600	\$3.00	0.00300	0.00500	0.00600	0.00300	
Aluminum, 6061 T6	61	0.098	400	\$3.00	0.00300	0.00500	0.00600	0.00200	
Brass, FC 360	51	0.308	800	\$5.00	0.00300	0.00600	0.00700	0.00400	
Brass, Naval 464	53	0.304	180	\$5.00	0.00200	0.00400	0.00500	0.00200	
Stainless, 303	33	0.287	125	\$3.50	0.00200	0.00200	0.00400	0.00200	
Stainless, 304 (A2)	34	0.287	75	\$3.50	0.00200	0.00200	0.00300	0.00150	
Stainless, 410	41	0.287	120	\$3.25	0.00300	0.00300	0.00400	0.00200	
Steel, 1040,1045	14	0.284	105	\$0.90	0.00200	0.00200	0.00300	0.00150	
Steel, 12L14	01	0.284	250	\$0.70	0.00300	0.00300	0.00600	0.00250	
Steel, 4140	24	0.284	110	\$1.50	0.00200	0.00250	0.00400	0.00150	

Fig. 6: Material Properties-SQL



## Building the Java Interfaces

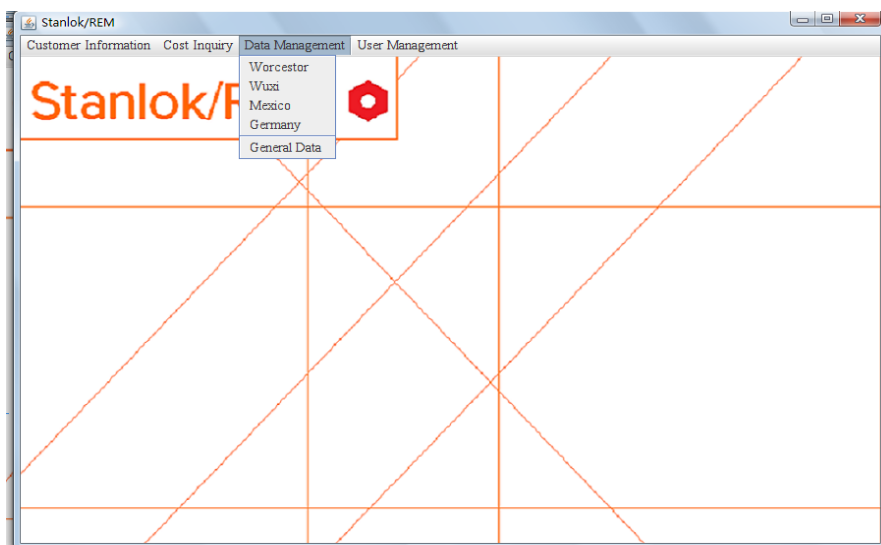
Next, we built a Java user interface. Initially, a user will see our “User Access” interface.

It contains a username and password input, and looks like this:



*Fig. 7: User Access Interface*

Once a user is granted access with a correct userID and password, he or she will see our “Main” interface. It is the heart of our program and contains tabs for “Customer Information”, “Cost Inquiry”, “Data Management”, and “User Management”. Each of these tabs and their related functions will be discussed shortly, but it should be noted that depending on the user’s access authority, they may not be able to use any or some of the tabs. Our “Main” page looks like this:



*Fig. 8: Main Interface*

The “Customer Information” interface can be seen below. It contains all of the inputs required to perform the cost estimation, and any customer information required for documentation and future review.

Fig. 9: Customer Information Input Interface

The “Cost Inquiry” tab contains five options, Worcester, Wuxi, Mexico, Germany, and Total, signifying the four example locations we chose to perform cost estimation, plus a “Total” simultaneous cost output for all 4 locations at once. Selecting one location, “Worcester” for example, will display the following interface:

Worcestor Cost Output

Quote 01-.7544-5013... Tool\$, Gage\$ ? 25

Total cost per 100

Cost Elements	Qty1	Qty2	Qty3	Qty4	Qty5
Quantity					
Material Qty Lbs					
Material Cost (\$C, 1 Bar, \$50)					
Manufacturing Cost (\$/C)					
Set Up, Lot Charges (\$/Lot)					
Plating, Finishing (\$/lot, \$/C)					
Heat Treating (\$/Lot, \$/C)					
Tooling, Gage Charges (\$/Lot)					
Total Cost \$/Lot or \$/100					
Gross Margin Target					
Sales Price \$/100, Calculated					

Sales Price

Sales Price \$/100, Actual					
Sales Price \$/Each, Actual					
Order Value					

Submit

Fig. 10: Cost Inquiry Interface

Next, by selecting the “Data Management” tab sub-menus, users will see an interface like the ones shown below, which allow for adding, modifying, or deleting any data located on the SQL server (for example, material prices or tool specifications, as markets or machines change). These operations can be performed for all four locations. This allows for up-to-date information for an accurate cost calculation.

Worcestor Data Management

Heat Data Plating Data Mechine Data Tool Data Exit

Worcestor Heat Treatment Data Manage

Search by Description: A194 Gr 2H Rc 24-35 ... Search

Code: 2H Description: 194 Gr 2H Rc 24-35

Min\_Cost: 120.00 Cost/Lb: 1.00000

Add Delete Save

Fig. 11: Worcestor Data Management Interface

The screenshot shows a software window titled "General Data Management" with three tabs: "Material", "Size/Rata", and "Exit". The "Material" tab is active. At the top, there is a "Material Type" dropdown menu set to "Steel, 1040,1045" and a "Search" button. Below this is a section titled "Material Properties" with fields for "Code" (14), "Type" (Steel, 1040,1045), "Density" (0.28400), and "SFPM" (105). The next section is "Cost/Lb in diffirent Sub-company" with fields for "Worcestor" (0.9), "Wuxi" (0.45), "Mexico" (0.9), and "Germany" (1.9). The final section is "Auto Lathe Feed Rates" with fields for "Cut Off" (0.0020), "Drill Center" (0.0020), "Drill Tap" (0.0030), and "Form" (0.0015). At the bottom, there are three buttons: "Add", "Delete", and "Save".

Fig. 12: General Data Management Interface

Finally, by selecting the “User Management” interface, a user can determine who can and cannot see any or all of the “Main” interface tabs. For example, giving finance employees or management access to “Data Management” only allows someone with specific knowledge of prices and rates to be in control of those features, whereas a company engineer performing an order quote for a customer only needs to be able to input the customer order (in the “Customer Information” interface) and read the outputted costs (in the “Information Inquiry” page), and only needs access to those two tabs. In this way, management not only can control who has access to the program, but *how* users will use the program. Keeping authority control at a high level like this will help to eliminate the “evolution” that occurs in many cost estimation systems. The “User Management” interface looks like this:



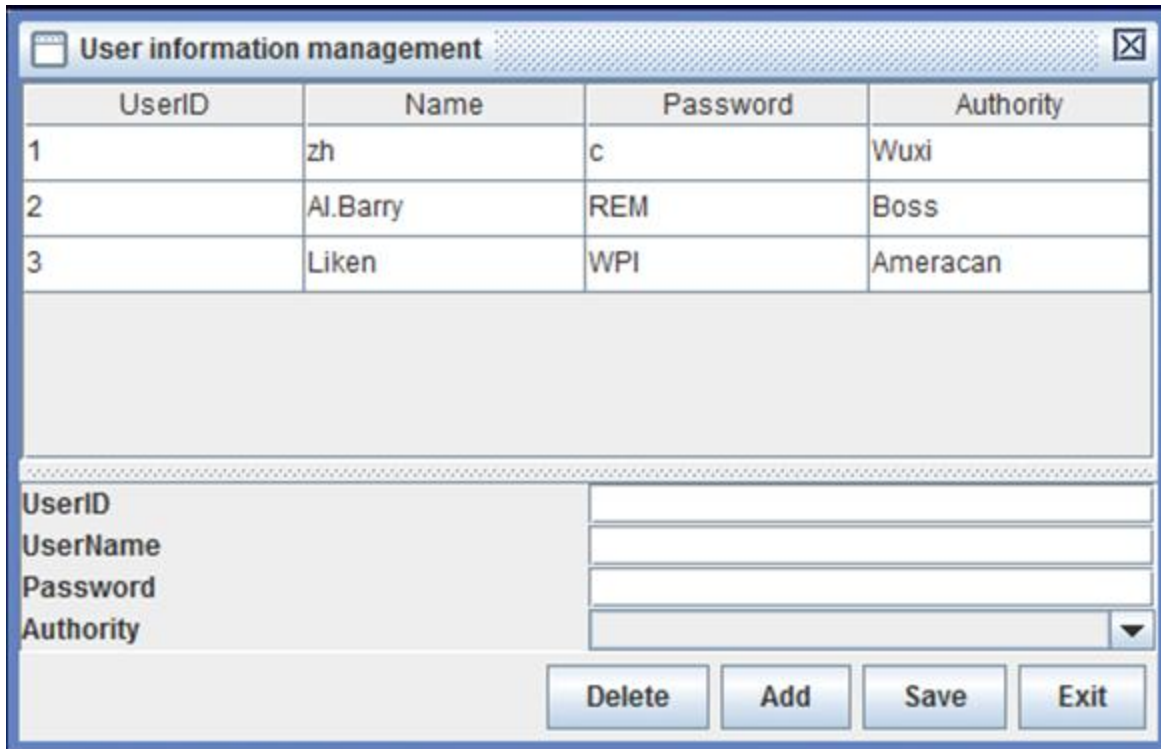


Fig. 3: User management Interface

Using Java’s JDBC functions

Now that we have both components of our final program, we must have a way to link the data in the SQL server to the calculator functions in the Java interface. Turning data into useable information is the driving force behind any cost estimation system. Using the Java language, we have been able to turn the previously discussed cost calculation equations into Java code, enabling functionality to our interfaces. While we cannot show *all* of the relations encoded into

our program, here is how material cost is calculated and the Java code that performs this operation:

$$M = \left(\frac{q}{p}\right)(ud)$$

Where: M = Total material cost

$q$  = quantity of parts requested

$p$  = number of pieces per bar

$u$  = lbs per bar

$d$  = U.S. dollars per lb

Corresponding Java code:

```
private double get_Material_Cost(double C0,double C1) {
    // TODO Auto-generated method stub
    double Mat1_LB = getElement("MaterialData",4,"MaterialType",vdata.get(8).toString());
    double F5=C1/C0;
    double F7=Math.floor(144/((Double.parseDouble(vdata.get(14).toString())
        +getElement("Material_Size_Data",5,"AF_Size",Double.parseDouble(vdata.get(13).toString()))
        +0.005)*1.03));
    double Lbs_Bar = F5*F7;
    if(C0>=100){
        return Math.max(Math.max(50,Mat1_LB*Lbs_Bar),Mat1_LB*C1)/(C0/100.0);
    }
    else{
        return Math.max(Math.max(50,Mat1_LB*Lbs_Bar),Mat1_LB*C1);
    }
}
```

# CONCLUSION

Our final product is a piece of software that is able to calculate the delivered costs at different locations of Stanlok/REM simultaneously, serving as a reference for management decision-making and cost quote tool for engineers and customers.

Structurally, we have created a database management system constructed with Microsoft SQL Server 2005. It contains all the data related to material, manufacturing, shipping and labor cost of four factories of Stanlok/REM (Worcester, Wuxi, Germany and Mexico). The interfaces are built and connected to the database by Java and its integrated database connectivity tools. Utilizing our software, all the engineers of Stanlok/REM have easy access to the inquire information about total delivered, material and manufacturing costs from anywhere in the world, which realizes information sharing in the whole company. Only users with proper authority have access to data management - adding, modifying and deleting the data in the database. The concrete calculating equations are concealed from the interfaces and need not to be taken into consideration by the users. The inquiry and data managing are simple operations, performed on the interfaces by clicking a button, choosing an item in the list-menus or typing in data in the pop-up dialog boxes.

The software accomplishes our goal of comparing product total delivered costs among all Stanlok/REM factories and is accessible to all of the company's manufacturing engineers and management from any location. In this way, the resources of Stanlok/REM are readily available to all manufacturing engineers and management within the global corporation, improving collaboration and communication between differing branches of the company.

This project proved more difficult than anticipated. It was simply “easier said than done”. Creating a simultaneous cost estimation program for four separate locations seemed like a simple job at first. Create a single cost calculation model and copy it four times, concluding with a side-by-side comparison of all four outputs. Job done, right?

Not so fast. Once we got into the specifics of the project, mainly how to build the calculator and the calculation functions themselves, our task list grew exponentially. Each equation used in the estimation process had to be analyzed and its relationships recorded, to be put into Java code form later on. The various data fields had to be reviewed and normalized so that they could be placed onto an online SQL server. And finally, the most difficult part of the project, learning Java code.

Prior to this project, only one team member had experience using or writing Java code. Similarly, only one team member had experience writing SQL. So while the other four members analyzed equations, created hundreds of relationships, and built flow charts to help aid in programming the functions, it was ultimately left up to those two teammates to actually do the programming, day in and day out, to complete the structure and functions we needed for our cost estimation.

But it is the feeling of accomplishment that made our project so much more than a computer program. We started with a seemingly simple problem, broke it down into pieces, analyzed each and every piece, made a phase’s →activities →tasks list, and formulated a technically sound engineering work plan from scratch. Utilizing the engineering process, along with many drawings, pictures, hand waving and even interpretive dance moves on occasion, we overcame our language and cultural differences as a team and built a simultaneous cost

estimation program that will improve collaboration globally within the Stanlok/REM partnership.

We cannot thank everyone involved with this project program enough. Thanks to the support of the Huazhong University of Science and Technology, the Worcester Polytechnic Institute, Stanlok/REM Wuxi Ltd. and Al Barry's generous sponsorship, and Professors Yiming Rong and Lingsong He, we have spent eight weeks in China, learning, growing, and understanding the level of expertise, passion, and commitment that will be required for us to succeed as engineering professionals. We have gained many friends along the way, and without the help and guidance of our HUST teammates, Tong Liuzhu, He Dan, Yang Xin and Zhang Chunjiang we would have never accomplished what we have. The Global Manufacturing and Collaborative Tools project was truly a collaborative effort by all those involved.

# BIBLIOGRAPHY

1. The Stanlok Group, “About Us”, <http://www.stanlok.com/aboutus.html>, accessed April 10, 2010.
2. Wuxi REM Co. Ltd., <http://www.remchina.cn/>, accessed April 10, 2010.
3. Maddocks AP, Dickens JG, Crawford AR, 2002, *Encouraging Lifelong Learning by means of a Web-based Personal and Professional Development Tool*, “The Skills, Attributes and Qualities of an Engineer”, ICEE 2002, UMIST, Manchester, <http://www.pble.ac.uk/pble-sd/skills-attrib-qual.pdf>, accessed April 23, 2010.
4. Microsoft, “Frequently Asked Questions about Remote Desktop”, <http://www.microsoft.com/windowsxp/using/mobility/rdfaq.msp>, accessed April 17, 2010.
5. Microsoft-TechNet, “How Terminal Services Work”, [http://technet.microsoft.com/en-us/library/cc755399\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc755399(WS.10).aspx), accessed April 17, 2010.
6. GoogleBlog, “Upload your files and access them anywhere with Google Docs”, <http://googleblog.blogspot.com/2010/01/upload-your-files-and-access-them.html>, accessed April 17, 2010.
7. Economy-Point.org, “Manufacturing Costs”, <http://www.economy-point.org/m/manufacturing-costs.html>, accessed April 23, 2010.
8. Holweg, Matthias. "The genealogy of lean production". *Journal of Operations Management* 25 (2): 420–437. 2007.
9. Korpela, Jukka, Lehmusvaara, Antti, Tuominen, Markku. “Customer service based design of the supply chain”. *International Journal of Production Economics*: 25-29. 2001.
10. The Access Web, “FAQ’s”, <http://www.mvps.org/access/>, accessed July 28, 2010.

11. CodeBetter.com, “Sharepoint is not a good development platform”,  
<http://codebetter.com/blogs/jeffrey.palermo/archive/2007/09/13/sharepoint-is-not-a-good-development-platform.aspx>, accessed August 3, 2010.
12. Microsoft Corporation, “Sharepoint benefits for developers”,  
<http://sharepoint.microsoft.com/en-us/product/benefits/developer/Pages/default.aspx>,  
accessed August 3 2010.
13. SQL Server Magazine, “SQL Server 2005 Management Tools”,  
<http://www.sqlmag.com/article/tsql3/sql-server-2005-management-tools.aspx>, accessed  
August 4, 2010.
14. Web.Archive.org, “FAQ about Oracle”,  
<http://web.archive.org/web/20080116210119/http://www.orafaq.com/faqora.htm#SCOTT>  
, accessed August 2, 2010.
15. Oracle Database, “Documentation Library”, <http://www.sysdba.de/oracle-dokumentation/11.1/>, accessed August 5, 2010.
16. Sun Microsystems, “White Paper-The Java Language Environment”,  
<http://java.sun.com/docs/white/langenv/Intro.doc1.html#943>, accessed August 5, 2010.
17. Sun Developer Network, “The History of Java Technology”,  
<http://www.java.com/en/javahistory/>, accessed August 5, 2010.
18. Eclipse.wiki, “Where did Eclipse come from?”,  
[http://wiki.eclipse.org/FAQ\\_Where\\_did\\_Eclipse\\_come\\_from%3F](http://wiki.eclipse.org/FAQ_Where_did_Eclipse_come_from%3F), accessed August 5,  
2010.
19. Eclipse.org, “Eclipse and Java For Total Beginners”,  
<http://www.eclipse.org/resources/resource.php?id=392>, accessed August 5, 2010.