

Fall 12-1-2004

E-Commerce Implementation for Dutch Precision Inc.

Daniel C. Mortenson
Dakota State University

Follow this and additional works at: <https://scholar.dsu.edu/theses>

Recommended Citation

Mortenson, Daniel C., "E-Commerce Implementation for Dutch Precision Inc." (2004). *Masters Theses*. 60.
<https://scholar.dsu.edu/theses/60>

This Thesis is brought to you for free and open access by Beadle Scholar. It has been accepted for inclusion in Masters Theses by an authorized administrator of Beadle Scholar. For more information, please contact repository@dsu.edu.

E-Commerce Implementation for Dutch Precision Inc.

Daniel C. Mortenson

A project submitted in partial fulfillment of the requirements for the

Master of Science in Information Systems

Dakota State University

2004



MSIS

PROJECT APPROVAL FORM

Student Name: Daniel C. Mortenson

Expected Graduation Date: December 30, 2004

Master's Project Title: E-Commerce Implementation for Dutch Precision Inc.

Date Project Plan Approved: May 2004

Date Project Coordinator Notified and Grade Submitted: _____

Approvals/Signatures:

Student: Daniel C. Mortenson

Date: 12/22/04

Faculty supervisor: Jepai Zhou

Date: 12/22/04

Committee member: Tom Farrell

Date: 12/22/04

Committee member: Michael L. Waldner

Date: 12/22/04

Abstract

The MSIS project at Dakota State University, "E-commerce implementation for Dutch Precision, Inc." began in May 2004 and was completed in December of the same year. The target company, Dutch Precision, Inc. (DPI), remanufactures brake calipers for automotive applications, regionally based near Tampa, Florida, and sells predominately to independent auto parts stores.

The DPI company website was established on a new static IP address using the DutchPrecision.com domain name. Three interrelated open-source code software applications were used; Apache web server, MySQL database, and PHP, a server-side scripting language.

Virus and networking difficulties initially impaired and refocused the project goals, but these were largely overcome. In the end, however, despite the development of a live website, concerns for security, low level long-term technical support, and financial factors drove the decision to establish the website with a qualified Internet Service provider.

Areas of student growth included networking systems and protocols, firewall and router configuration, programming, and database integration.

Table of Contents

Project Approval Form	ii
Project Abstract	iii
Table of Contents	iv
List of Tables	vi
List of Figures	vii
Introduction	1
Project Objectives	3
Implementation and Development	6
Analysis	6
The Business	6
The Computer Network	14
Computer Hardware and Software	15
Computer Hardware.....	15
Upgrading Hardware.....	17
Computer Software	19
Upgrading Software.....	20
Computer Security.....	22
Analysis Summary	25
Website Design	31
Design	31
Software	38
Imaging	39
Database Development	41
Data Alignment	41

Database Design.....	44
Findings and Discussions	47
Apache	47
Php	49
MySQL	49
Government Contracts.....	50
Conclusions and Future Work	53
References	54

Appendices

Appendix A:

Work Breakdown Structure	A-1
Gantt Project Event Schedule.....	A-2
Appendix B: Website coding.....	B-1
Appendix C: Database coding	C-1
Appendix D: Web Design Imagery	D-1
Appendix E: Pictures of DPI	E-1

List of Tables

Table D-1: Website Image Designs.....	D-1
Table D-2 Website Image Designs Cont.	D-2
Table E-1: Pictures of Dutch Precision, Inc.	E-1

List of Figures

Figure 1: A DPI Logo.....	1
Figure 2: How a Disk Brake Works.....	9
Figure 3: Diagram of Business and Data Flows	13
Figure 4: DPI Network as First Encountered.....	16
Figure 5: DPI Network – Final Implementation.....	19
Figure 6: Linksys BEFSR-41 General Settings.....	23
Figure 7: Router Filters Example	24
Figure 8: Router MAC Address Filtering	24
Figure 9: Screen capture – DPI Website Index.php	34
Figure10: Screen capture – DPI Website Products.php.....	34
Figure 11: Screen capture – DPI Website PartsLookup.php.....	35
Figure 12: Screen capture – DPI Website About DPI.php	35
Figure 13: Screen capture – Sub Parts Lookup – html lists.....	36
Figure 14: Screen capture – Browser-based database Admin	36
Figure 15: Screen capture – Browser Data Entry.....	37
Figure 16: Screen capture – Database Query Filters	37
Figure 17: Screen capture – Sample Query Results	38
Figure 18: DPI Business Card.....	41
Figure 19: Recreated Logo	41
Figure 20: Sample View of DPI Data Spreadsheet.....	42
Figure 21: Design Structure and Cardinalities of Web Based Caliper Reference Database.....	45
Figure 22: Number of Hostnames and Active Internet Sites.....	47
Figure 23: Market Share For Top Servers Across All Domains	47
Figure 24: PHP growth.....	49

Introduction

The proposed project was to create a website or web presence for Dutch Precision, Inc. (DPI), an automobile brake caliper remanufacturing company located in Port Richey, Florida. Through this website, DPI desired to better serve their current customers by providing an online parts number reference, attract new customers through online availability, and place the company in a more favorable position to bid on government contracts by creating greater accessibility to their identity. The web site would need to be scalable and compatible with future plans for secure, transactional, e-commerce.

The project included a general analysis of the current business and information systems used by DPI, designed to familiarize the graduate student with the network infrastructure, database environment, and security systems and policies, and to crystallize the expected adjustments or improvements needed to implement the project.

Significant network infrastructure issues were encountered due to aging and varied computer hardware, protocols, software, viruses, and a combination of limited resources (financial, technical and temporal). Each was addressed with varying degrees of success.

A new website was designed and partially implemented on a new Dutch Precision web server (www.dutchprecision.com) using three open-sourced

software applications. MySQL (4.0.20a) was the selected database program, interfaced with PHP (4.3.9), a server-side scripting language, and Apache 2 (2.0.50) web server software.

In addition to the benefits realized by Dutch Precision Inc., the graduate student advanced personal research and understanding of these and other relevant topic areas: CSS (cascading style sheets), PHP, MySQL, Apache, image manipulation, network protocols, router and hub configuration, firewalls, PC Anywhere (remote access control software), domain name issues, port vulnerabilities, virus protection, and data manipulation.

Project Objectives

This project had six general and interrelated objectives. They are listed here, and a broader definition of each follows:

1. Perform a business and IT systems analysis.
2. Upgrade the existing hardware and software systems.
3. Design a website with E-commerce compatibility.
4. Design a parts database accessible from the website.
5. Use Php as the interconnecting scripting language.
6. Enhance the knowledge and experience of the graduate student.

The first objective of this project was to gather data about DPI business processes and to effectively analyze the current usage of computer technology in the company. This resulted in recommendations for changes thought to improve the existing business and computer systems used throughout the company. Understanding the business aided the graduate student in designing the company website.

The second objective was to make upgrades as necessary to the existing computer network and hardware to incorporate the new use of a web server and its related applications. This included upgrading equipment, where necessary or permissible, and doing general network maintenance as revealed in the system analysis, and ultimately the installation of the web server, database, and server-side scripting language.

Objective number three was to design the DPI website in a test environment, targeting current customer needs. The underlying structure was to be designed and approved, and then two or three layout designs proffered with one, ultimately, approved.

The website design needed to be scalable, anticipating more complex features added in the future. To do this a three-tiered structure was envisioned. The first tier was to be accessible by customers and the general public alike, providing information about DPI, its products, and contact information. The second tier will require registration and password authorization, designed to serve existing customers with order placement and account status. Then the third tier will be a store-front for retail buyers. This last tier will involve coordination of ordering, payments via credit card, and shipping - both of the new calipers and any returned caliper core.

Project deliverables for the third objective include a website structure diagram, screen shots of each web page, and a functioning version of the first tier of the website offline.

The fourth objective was to design a parts reference database accessible on the website. This was to include a structural diagram and a functioning version of the database tables and query forms and would require the migration and population transfer of real DPI parts data. The aggregation of DPI data proved

to be a formidable task, and the relative success of this objective will be discussed at length in this paper.

The fifth and objective was to create an integration or communication between the web server and the database using Php scripting. The code itself will be one deliverable, along with screen shots or live demonstration of functioning integration in a test environment.

The sixth and final objective was to enhance the knowledge and experience of the graduate student with regards to information systems and their application in real world environments. The graduate student specialized in the Networking emphasis of the MSIS program. As such, the exposure to E-commerce, programming, and database design proved to be enlightening and invaluable to his overall understanding of Information Systems as a whole.

Implementation and Development

Business and IT Systems Analysis

The Business: To achieve the first objective, the graduate student gathered facts through observation, employee interviews, and then



Figure 1: A DPI Logo

organized these findings through systems modeling and written statements. This analysis was performed in May 2004, with observation and interviews taking place during the ten day on-sight visit to Florida, and the final modeling occurred after that time.

The purpose of the analysis was to reinforce understanding of the basic business data and resource flows within the organization prior to the inclusion of any E-commerce systems.

Dutch Precision, Inc. (DPI) remanufactures brake calipers to original equipment manufacturer (OEM) standards for foreign and domestic cars, trucks, and trailers. Located at 5914-A Dasher Court in Port Richey, Florida, DPI sells its calipers to retail automotive stores mostly in south central Florida and along the Florida coast of the Gulf of Mexico. It also sells to a warehousing system located in Miami. The business has seen progressive expansion of its geographical range of customers.

Owner and manager, Barry Van Wechel, started the business in 1997, developing a local customer following, and then bought out Coastal Caliper Remanufacturing Inc., a regional competitor, in 2000. The business grew moderately, but consistently, and he wanted to begin the development of an E-commerce capable web site for DPI. This desire was based on a general perception that selling on the Internet is the "wave of the future", and that opportunity for growth and increased market share are available through electronic means to those who develop a viable website. Further discussion later in this report supports this general assertion.

DPI resides in a single, leased building, utilizing 8064 square feet. The building has an upper level covering 7200 square feet that is owned or leased by other private individuals. Estimated floor space divisions are as follows (given in square feet):

- | | |
|---------------------------|----------------------------|
| - Office space – 864 | - Breakdown/Cleaning -1000 |
| - Caliper Assembly - 1400 | - Distribution 1000 |
| - Core Storage – 1200 | - Parts 800 |
| - Caliper Storage -1600 | - Other usage - remainder |

Dutch Precision has more than doubled in physical size and product sales in its seven year existence. The increased revenues have been used to expand to its current facility and to increase caliper core stock.

DPI has garnered a reputation for providing quality calipers with a willingness to accommodate compressed order schedules and unique inventory items. 98% of its sales are in Business to Business relationships. Their primary customers are automotive parts stores in Central, Southern, and Western Florida. DPI will sell to individual 'walk-in' customers, but does no marketing to attract such customers.

DPI is relatively small, having just four full-time and three part-time employees, and a single salesman who works on commission.

DPI business functions can be divided into two major segments – Sales and Remanufacturing. The Sales system includes all customer relations, order management, and accounting, while Remanufacturing receives supplies, produces products, and distributes completed brake calipers.

As part of the business systems analysis, it was essential to understand the product being produced, the vocabulary of the caliper business, discovering how the system works, and then to follow an actual caliper through the system from start to finish.

A brake caliper is a c-shaped device on an automobile used with disc brakes. When a driver presses on the brake pedal, the caliper piston squeezes the brake pads against the rotor and causes the car to slow down or stop. Brake pads are

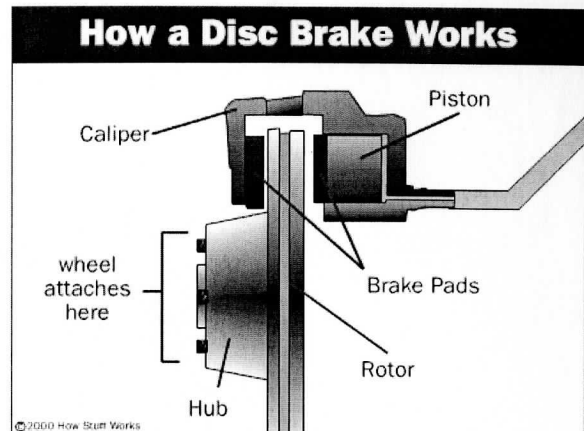


Figure 2: How A Disc Brake Works

mounted to the calipers, which "float" next to the rotor or brake disc. The caliper ensures that the brake pads exert even pressure on the disc (GlobalAutoparts.com, Gemini.Goodyear.com).

Automobiles are classified by their year of production, their model or type, and by their make or manufacturer. There are also sub-applications of many models. Websites for auto parts retailers list anywhere from 40 to 82 makes of cars, with each manufacturer producing five model types, on average, but with ranges from one to sixteen models. Of course some of these models do not use brake calipers. Others use them on the forward wheels or back wheels or both. Often, but not always, the left and right side calipers are interchangeable.

Dutch Precision keeps records on 95 different makes of cars and trucks, with listings for 747 models. When divided into ranges of years and left and right position, the separate caliper applications come to 7062, and increasing each year.

From the perspective of Dutch Precision, the term "caliper" is used to define a finished product that has been produced, placed into inventory, and is ready to be sold. A "core" or a "caliper core" is a returned or newly purchased used caliper that is not reconditioned or is in an incomplete state of reconditioning.

Calipers are generally sold in left and right pairs, though their identification numbers are unique. Calipers can be unloaded, loaded, or semi-loaded, and they can have brake pads installed (loaded) or not. More commonly, brake pads are sold separately by the individual auto parts stores.

To rightly understand caliper remanufacturing, one must understand that returned caliper cores are at the heart of Dutch Precision's business. A core, to offer a definition, is a used caliper body, or casting, that is not in functional order. Cores are grimy, unsightly, and worth little even as scrap metal, generally, due to their cast-metal composition.

When DPI has to purchase a core to restore it and resell it, there is typically zero profit and, even more likely, significant loss per item. Each time, however, that a core, or its equivalent casting, is returned, in exchange for a customer deposit, and reused the initial investment is reduced by half. Cores in high demand may be recycled hundreds of times, effectively reducing the cost of that caliper core to a few cents.

Assessing the value of a non-refurbished caliper core, consequently, becomes somewhat of a difficult task. A specific core may rightly have an assessed value of a few pennies or as much as \$75.00, based upon customer deposit, while its resale value ranges from, say, \$12.00 to \$50.00 depending on the vehicle application. Such a valuation “game” is not unique to DPI and the debate between the Internal Revenue Service and the business community will not be resolved in this paper. The purpose of this discussion, however, is simply to illuminate the motivation for businesses like DPI to have a limited inventory of cores – preferring to recycle returns perpetually– and yet facing the reality that cores are not always returned, are rare, in diminishing demand, or are damaged beyond repair, necessitating another initial purchase.

That being said, the matter of caliper core returns and valuation is essentially the only thing unusual or difficult about DPI’s business. In every other respect its business practices are well defined and common – matters of product, customers, sales, purchase orders, accounts payable and receivable, delivery/distribution, and internal matters of employees, payroll, benefits and time sheets.

The graduate student also interviewed owner Barry Van Wechel extensively, and met with each employee during the on-sight visit. This was essential to

understanding caliper numbers, product labeling, delivery routes, weekly production routines, the billing process, and caliper remanufacturing steps.

Dutch Precision's management situation, both for long-term vision and for daily operations, is inextricably linked to its owner, Mr. Van Wechel. Van Wechel designed and has performed every operation and task of the business, without exception. He still is involved in every aspect of the production process, depending on the weekly product demand, though much of his time is spent coordinating the activities of his workers, resolving technical questions, dealing with customers, and maintaining the data systems.

Van Wechel also actively seeks to train and empower his employees, though he has had limited long term success training managers capable of fully replacing himself.

Van Wechel's management style is pragmatic, calculating and decisive. The formal systems and record keeping that are in place have been developed to increase efficiencies, comply with legal regulations, and reduce complexities in training employees, but they reflect the personality of Van Wechel and their perceived value to him.

The business major processes and flow of data are diagramed below in figure 3. Not pictured are the details of parts suppliers and the attending accounts payables incumbent therein. Also, general entity relationship symbols overlap

with and more closely reflect data flow modeling techniques. For a better understanding of the details of data cardinalities and data joins refer to figure X on page xc.

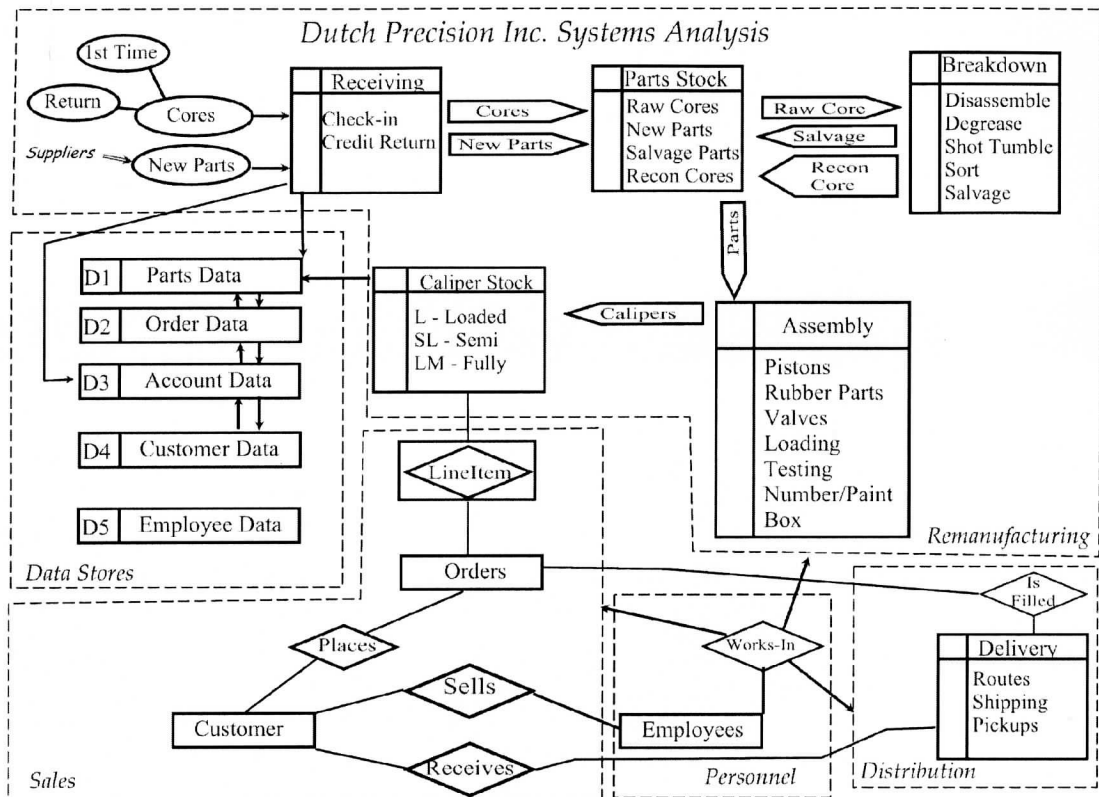


Figure 3: Diagram of business and data flows

In the estimation of the graduate student the greatest strength of the Dutch Precision business system is Mr. Van Wechel himself – his strong work ethic, unwavering commitment to product quality, and his firm grasp of the details of caliper remanufacturing. However, the strength of DPI is also its greatest vulnerability. This analyst feels that it would be difficult for DPI to remain viable as a business without having another individual or dedicated team to

replace Mr. Van Wechel. This is not to fault any of the current business practices, for they appear to be sound, but Van Wechel will have to duplicate himself with increasingly independent managers in order to see growth similar to that of the first seven years.

The Computer Network

A second, but parallel, analysis was made of the computer network and information systems being used by Dutch Precision.

On May 8, 2004, two days prior to the scheduled beginning of this project, the DPI computer network, simple as it is, was attacked and rendered useless by a computer virus. The failure is actually believed to have been the result of malicious code and several opportunistic Trojan viruses that were exploiting a poorly protected DSL connection to the Internet. The graduate student had a ten day on-location schedule in Florida, but arrived to disarray and a non-functioning computer network.

In dire need, Dutch Precision had hired external technical support from a local business, Ascom Computer Specialists. When the graduate student arrived, an Ascom employee was in the process of replacing and reformatting the hard drive, upgrading the drive from a single hard drive with a second removable back-up drive, to a mirrored, Raid-Level 1 drive (maintaining the removable

drive too). He reinstalled Windows 2000 Professional, and removed a non-WEP-enabled wireless router.

Before continuing the sequence of events, an explanation of the DPI network is in order to fully understand the network systems in place.

Upgrading Hardware and Software

The next objective was to upgrade hardware and software as needed.

Computer Hardware:

Dutch Precision has five PC computers operating on an Ethernet peer-to-peer network, arranged in a star topology. Four of the PCs are desktop models with mini tower housings, and one is a personal laptop. Three of the desktops are custom built 'FireNet' brand using Intel Pentium IV, 2 GHz processors. Each has 512 Mb of RAM. The other desktop computer is a much older Compaq Impressario 5140, with just 96 Mb RAM, and a Pentium II-processor.

One computer functions as a data file server, running the Windows 2000 Professional operating system. This computer now is also running the Apache web server and MySQL database applications. The laptop is running Windows XP, and the other three stations have Windows 98 SE. The computers were generally identified as 1) server, 2) front office, 3) middle office, 4) shop, and 5) laptop. A sixth computer is not on the network, but is dedicated to

communications via dial-up modem with the United Parcel Service. It is running Windows 95 and has a dedicated label printer.

The network utilizes Category-5 STP cabling with RJ-45 connectors attached to standard 10/100 Ethernet network interface cards (NIC). As illustrated in figure 4 below, the newly amended network (by Ascom technicians) used an eight port Netware brand hub to connect the shop, laptop, and two office computers to the file server via a dedicated LAN-side NIC. A second NIC on the server was dedicated to WAN connectivity with a direct connection to the DSL modem. The graduate student believes that a measure of security was to be enforced using separate network protocols with each NIC, though he was not, in fact, told the details of Ascom's intentions.

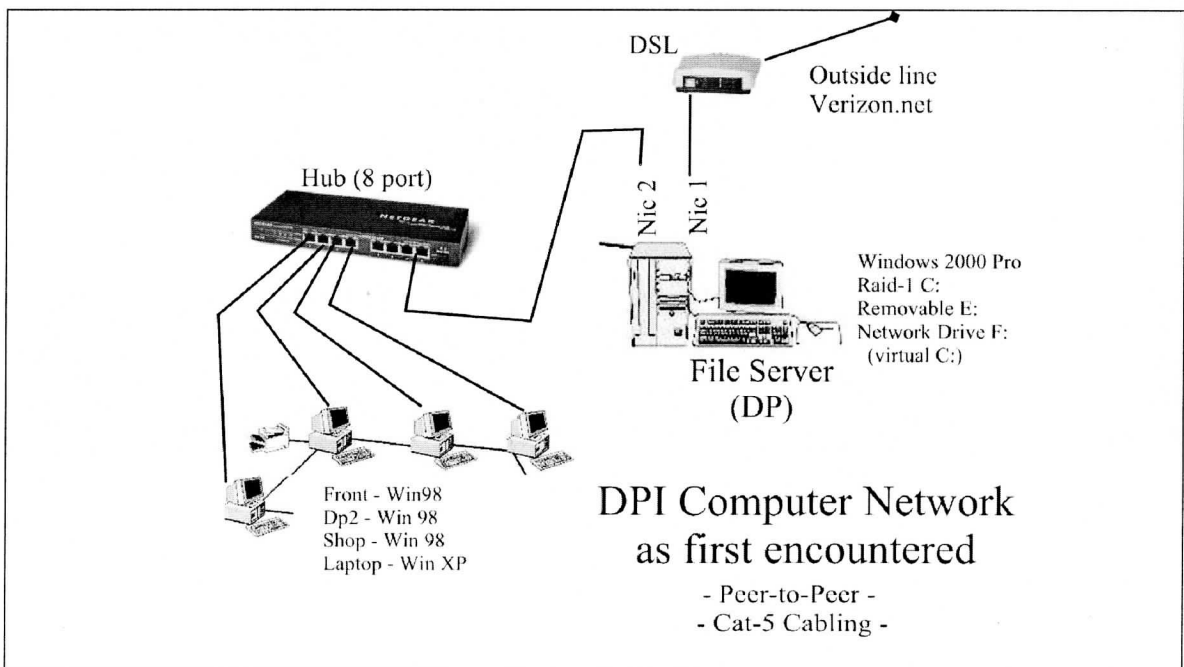


Figure 4: DPI Network as First Encountered

It is presumed that Ascom's plan was to leverage the differences between Microsoft NetBEUI LAN protocol and the TCP/IP protocols used on the WAN side. As such, it is possible to remove all TCP bindings from the LAN side NIC, effectively isolating the LAN from the WAN, and likewise, remove all NetBEUI and IPX protocols from the WAN side NIC. And, while the famous Dr. Seuss would likely have enjoyed this last paragraph, it is a fairly effective means of isolating a LAN from a WAN. However, it eliminates any Internet connectivity by computers other than the file server, and it leaves the server open to unwarranted vulnerabilities from external forces. As it happens, a "No Internet" policy was already in place for office and shop computers, but the laptop needed to access external email accounts and the Internet. And so a different solution was advanced.

Upgrading Hardware:

The network scheme developed and adopted is pictured in figure 5. It places a Linksys BESFR-41 router between the DSL modem and the rest of the network, allowing basic firewall control of ports, protocols, IPs and MAC addresses. The router is configured to dynamically assign IP addresses to all network devices except the server which has a static IP.

As simple as this may sound, the installation of the router was a unique learning experience in itself. Because finances were short, the student sought to purchase a used router through the online auction service, Ebay.com. He successfully found, at considerable savings, a Linksys – BESFR-11, which is a single port router, very similar to the BESFR-41 ultimately used. The BESFR-11 was purchased and delivered and installed (all via telephone communication), and the single port was connected to the Netware hub for distribution throughout the network. This implementation worked intermittently. Everything seemed to be configured properly, and would work, only to stop working a few hours later. Neither Mr. Van Wechel nor the graduate student had enough experience to be sure that the right procedures were being followed, so considerable time was spent setting and resetting the router – testing it with dynamically assigned IP addresses, and with static addresses, and with Mac address cloning, and in consultation with the Verizon DSL service. Ultimately nothing worked for more than a few hours, and decision was made to replace the router. The new router was installed, configured once and continues to work flawlessly. A lesson in true cost savings was learned.

Dutch Precision already had a DSL connection and modem through Verizon DSL services; however it had a dynamic IP address. It was necessary to get a static IP address for DPI to maintain consistent access to the web server. The

graduate student arranged for the static address, 66.14.118.45, and he registered the dutchprecision.com domain name on May 14, using the services of Register.com.

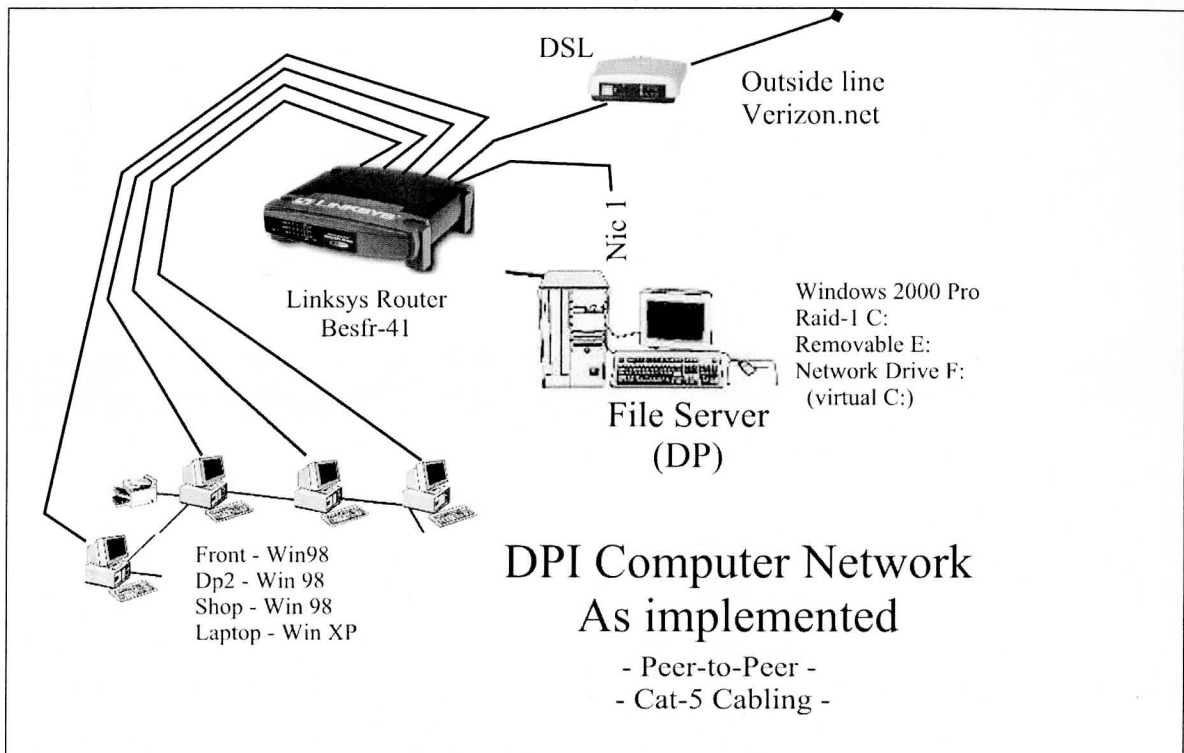


Figure 5: DPI Network – Final Implementation

Computer Software:

Software applications used by Dutch Precision were few in number, but central to business operations. DPI used MS-Word and MS-Excel in particular, and a DOS based, database program known as the Advantage Accounting System, developed by Roundtable Software (RTS) and supported initially by Armor Systems Consulting of Houston, Texas. DPI referred to it as "Armor". Armor systems went out of business and the software was unsupported for a

number years. Since then, limited support has been adopted by a third party company, with which there is no current relationship with DPI.

Armor is the key application in DPI's computer operations. It provides access to all transactional and inventory records of the company. It did not, however, provide general caliper reference data needed by Mr. Van Wechel. This data was stored in self-created and maintained Excel workbooks.

Prior to reformatting the server by Ascom, DPI was nearly functioning properly. The Armor database was functioning from all work stations, but some computers were not accessing shared printers correctly. When Ascom was dismissed (to be further explained) the front and middle offices were able to communicate, but the database was not accessible on the network. Armor was temporarily running from a single station in the front office.

Upgrading Software:

The graduate student and Mr. Van Wechel were able to configure the bat files on each computer on the network, properly share the network drive on the server, and get the Armor software program working. All of this was done via telephone calls, with the graduate student directing the activity performed by Mr. Van Wechel. A key factor in resolving the data anomalies lay in removing any local versions of the Armor program that remained on these other computers. Once the program was isolated to the shared drive, it functioned

accurately, and has been working properly since that time. The original printer problems were also resolved. And finally, via telephone conversation, Sygate Personal Firewall was installed and given basic configuration. Currently all data and print functions are working properly throughout the network.

With the system finally in place, the graduate student was able to install Symantec PCAnywhere 11.0 on his local computer, and the host was installed on DPI's server. The Linksys router and the Sygate firewall were set to recognize and allow the PCAnywhere connection, which uses a symmetrically encrypted password key system. This too was a new experience for the graduate student.

The next step was to download and install the Apache web server. Version 2.0 of this open source-code program has been in successful use long enough to be considered stable, and it was chosen. The installation process, while new to the student, was well documented, and the configuration file (`httpd.conf`) was set without difficulty. A few of the key items were to direct the server to the proper root access files for web documents, and to adjust the listening ports. Most of the default settings in the `httpd.conf` file are set for impose high security, and conscious effort must be taken to reduce these levels.

A similarly slow and careful installation process was followed for PHP and for MySQL without incident, and no other new software was introduced into the system.

Computer Security:

The security measures in place for the DPI network involve technology, policy, and employee training. The Linksys router employs network address translation (NAT) to help disguise or spoof the IP addresses of computers on the network, and the router has filters set to block standard Microsoft LAN communications ports 135-139 and port 445, as well as the ports exploited by several known security vulnerabilities (see figures 6 and 7).

The router is also configured to disallow internal access to the Internet for the specific MAC addresses of the shop and two office computers. This was deemed preferable to excluding their IP ranges, so that the laptop IP could be dynamically assigned without excluding access to the Internet.

As mentioned previously, the server also has the Sygate Personal Firewall installed. This software application monitors computer port activity, keeps logs, and has the ability to lock-out Denial of Service attacks. It also is not yet fully understood by either the graduate student or Mr. Van Wechel, and it's security settings, while implemented, have not been fully optimized.

Technologically, the final level of security involves the Symantec Anti-Virus software. The virus definition files are updated at least weekly, and a regular schedule scans all computer files with the current definitions in place.

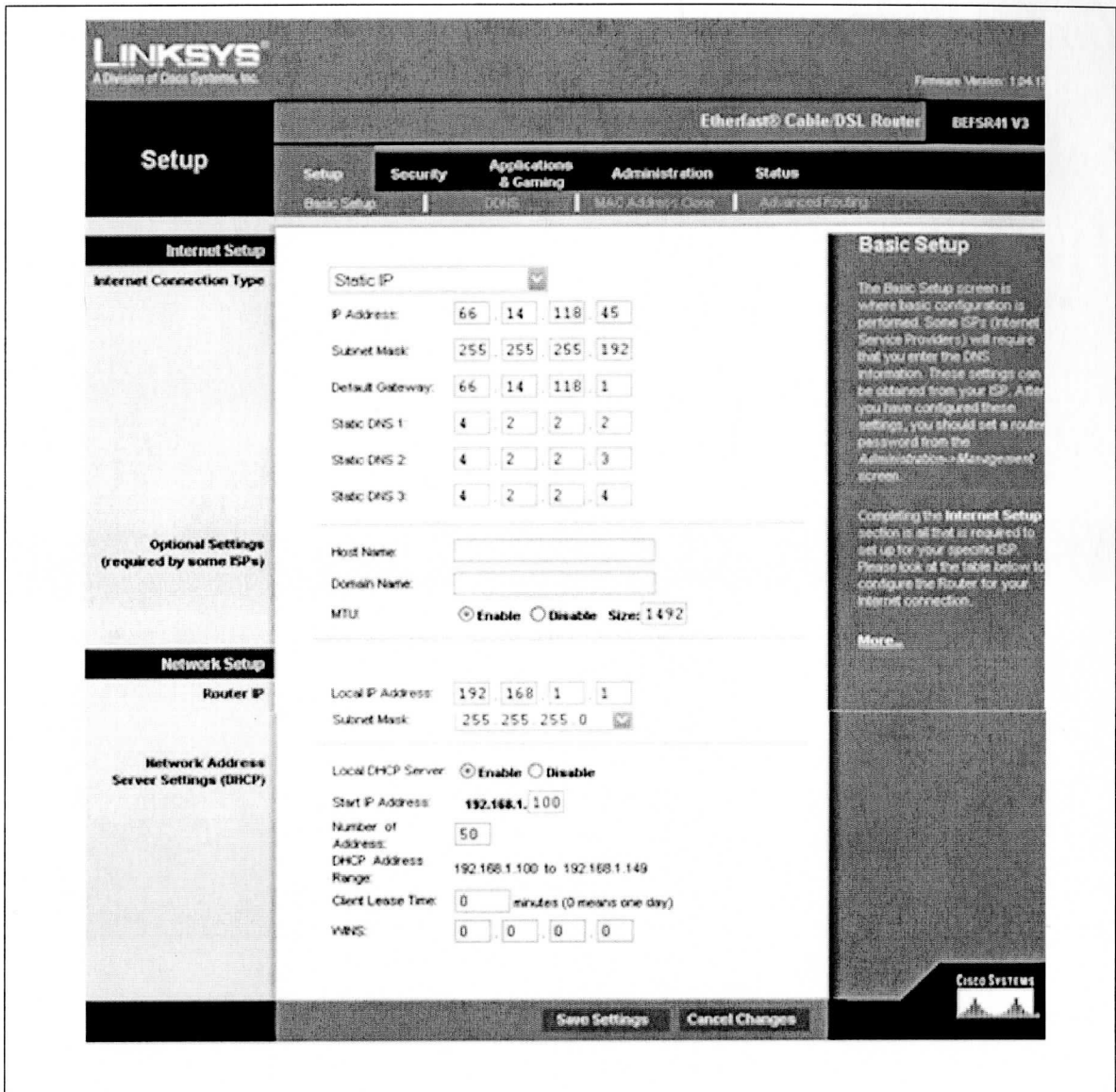


Figure 6: Linksys BEFSR-41 General Settings

NUM	Start	End
1:	192.168.1.0	to 0
2:	192.168.1.0	to 0
3:	192.168.1.0	to 0
4:	192.168.1.0	to 0
5:	192.168.1.0	to 0

NUM	Protocol	Start	End
1:	Both	135	to 139
2:	Both	445	to 445
3:	Both	1948	to 1948
4:	TCP	6667	to 6667
5:	Both	0	to 0

Block Anonymous Internet Requests: Enabled Disabled

Filter Multicast: Enabled Disabled

Filter Internet NAT Redirection: Enabled Disabled

Figure 7: Router Filters Example

Filtered MAC Address: 1~10

mac 1:	0080AD800B7F
mac 2:	0040D001E570
mac 3:	0040F46CBE00
mac 4:	0
mac 5:	0
mac 6:	0
mac 7:	0
mac 8:	0
mac 9:	0
mac 10:	0

Internet

Figure 8: Router MAC Address Filtering

In addition to technological security, Dutch Precision has implemented a few basic computer use policies and enacted employee use and policy training. First and foremost – Internet access is only available from the server and from the laptop, and then only by Mr. Van Wechel or persons having his per-session permission. File downloads, virus updates, and any other system maintenance are performed by Mr. Van Wechel. Employees may use the office and shop computers for DPI business only.

User passwords authenticate access to the network, though there is no requirement that these be changed regularly or be strong. Generally speaking, the current password system will only deter physical access to the network during an after-hour break-in to the facility. The Armor database program, however, requires and logs a separate password to gain access.

Analysis Summary:

Dutch Precision has business practices and a computer network basically adequate to perform its current operations, as indicated by its growth, though it has several inefficient practices that could be enhanced with integrated computer technology. Purchase order data, for example, is duplicated or reinterpreted at several stations, opening opportunity for errors or lost information.

When orders are received by telephone or fax, the information is entered into the Armor database system, generating accurate inventory and billing

information. This is then printed and delivered to assembly and distribution workers. Orders may be in inventory and pulled for the order from the shelves, while other line items were produced in inventory in name only, and still have to be produced in reality. Yet this is all identified by employees writing on the printed forms and physically distributed.

Labels, too, are not generated directly through the database program, but need to be identified visually and printed in a separate labeling program. The labels are often modified with A-1 Cardone numbers, or with a second label with graphics to conform to individual retailer needs. Much of this activity is coordinated by human interpretation of a printed purchase order.

A similar process is used to construct delivery route pallets, with shipping data entered yet another time for the DPI trucks, or in the UPS shipping system.

Ideally a single production database would generate each necessary step in this process, produce labels, and combine orders into logical delivery routes. It would also be helpful to identify calipers by computer scanned numbers or bar codes – but no such system is in place. Thus DPI functions smoothly through known and well rehearsed processes, but they are prone to human error and dependant upon well trained workers who understand the system.

The computer set up, also, is inadequate in its present configuration to maintain an e-commerce website. The graduate student recommended that a

separate web server be established that is isolated from the system and from other applications used for business operations.

As DPI grows, it will likely hire additional employees. More rigorous password policy will provide increased security and is inexpensive to implement. To do this efficiently, it is recommended that the server be upgraded from Windows 2000 Professional to Server software better suited to applications and domain level security controls.

These recommendations would be best implemented by a trained IT employee, or through a well established IT service provider. If that option is not economically viable, then it is here recommended that E-commerce activity be moved offsite to an E-commerce capable Internet service provider. The graduate student feels that this is the best, fastest, and most secure way for DPI to gain access to Internet sales.

A small company, like DPI, with limited technical, financial, and time resources, cannot reasonably compete with an Internet Host provider maintained by IT professionals. Such a provider can maintain more effective firewalls, server security, data backup, and other cutting edge technology while distributing those costs evenly to many customers. A small business must bear a much larger share of the technology costs if they choose to do everything "in-house".

For example, Dutch Precision pays approximately \$89.00 per month to maintain a static IP DSL line to their business. They could save about \$30.00 per month by switching back to a dynamic IP address which is a level of service sufficient for DPI's internet demands if they do not maintain a web server.

That moderate savings alone would purchase an E-commerce capable website from an Internet service provider. For between \$25.00 and \$35.00 per month from several different reputable Internet Service Providers, such as Register.com or Globat.com , a small business can have complete storefront web sales, including Secure Sockets Layer transactions credit cards, multiple email account, 24-hour professional IT service, and graphical interfaces that allow inexperienced persons to be successful online. The enormity of this value was brought home with significant emphasis to the graduate student throughout this project.

Never-the-less, the conclusion to abandon an E-commerce implementation did not yet occur, and the sequence of events now continues.

Ascom Computer Specialists had worked with DPI in the past, even helping to set up the network, though entirely through a former employee who was no longer with the firm. The Armor database needed to be installed on a shared network drive partition on the server with a drive letter name other than [c:] to distinguish it from the local hard drives of other computers on the network. The

network drive letter had always been [f:], and, as just mentioned, it had been a separate drive partition.

Batch files (bat) were configured on each remote computer to run the local installation of Armor, and then to orient all data storage and retrieval to the f-drive, and to adjust shared printer devices to various locations on the network. This careful configuration was essential, because Armor would not produce error codes or warnings when data files were improperly stored on a local hard drive, causing a progressive build-up of data anomalies that were difficult to repair. Previous experience had so taught.

The new Ascom employee was competent with Windows computer networks as a whole, but was inexperienced with DOS commands and with Armor in particular. When he formatted the new server hard drive, he created a virtual drive, named F:, and not an actual partition of the drive. In the end this proved to be an acceptable practice, with the C: and F: drives on the server being different in name only, but actually being the same physical space. This confused the Ascom technician, unfortunately, when it came to properly setting up share permissions, which behaved differently than for a drive partition. And to add stress to the environment, the technician was in great demand from larger and more influential customers at the same time – the local Walmart in particular. Mr. Van Wechel knew the importance of properly configuring the

server and related batch files, but was not readily able to explain the details to the Ascom technician. As such, the network repair at DPI proved to be more time consuming than was anticipated and was not getting done in a timely manner. The Ascom employee actually took the server hard-drive away from the DPI facility over a weekend, but failed to get it working, and kept the drive for several days. This did not help DPI's situation or Van Wechel's disposition.

In frustration, Ascom was dismissed, leaving a non-functioning network to Mr. Van Wechel and the graduate student. The resulting renewed network has already been detailed in the previous section, but the impact on project implementation was significant. First, while general observation and systems analysis was accomplished during the Florida visit, much more time was spent working on the network infrastructure than was expected, regardless of how appropriate this was to the student's training. As such, the plan to install the Apache server, MySQL database and Php did not happen at that time. These processes, and more network administration, had to be implemented remotely using Symantec pcAnywhere (11.0) software.

This is perhaps a good time to note the availability of the project work Breakdown Structure and Gantt chart in Appendix A. The increase in network processes necessitated a restructuring of expectations in other areas, particularly in the extent of implementation of e-commerce applications. And the realization

of his vulnerability to viruses and Trojans also caused Mr. Van Wechel to view his e-commerce goals with new skepticism.

Website Design:

The objective of creating an E-commerce capable website involved the following factors:

1. Defining the goals of the website, both short and long term
2. Understanding and matching the existing computer infrastructure
3. Preparing the software applications
4. Designing site and page layouts
5. Testing and implementing the new site
6. Maintaining the site with minimal technical support

Design: One major goal for the website was to provide an online database resource for current customers to lookup parts numbers. A-1 Cardone is a large national supplier of auto parts in the United States, including brake calipers. As such, their parts numbers are widely used, and retailer stores will often cross-reference DPI caliper numbers with A-1 caliper numbers.

In the original project plan it was proposed to provide current business customers the option to place their daily or weekly orders online, accessing their account information and the status of caliper inventories. This expectation was altered as more information was learned about Armor, the database presently being used by DPI, as security vulnerabilities were exposed, and as Mr. Van

Wechel reassessed the potentially negative impact of customer perceptions of low inventories.

One of the reasons DPI has gained market share is due to its ability to fill unusual or rushed caliper requests with rapid turn around time. If DPI knows that it has a core and can produce the caliper, the sale goes forward before the item is in inventory. And that is why an online inventory list may be counter productive, suggesting to a customer that DPI does not have the caliper to sell. The core inventory is not in the database as an individual casting, merely as part of a bin or group of like castings, perhaps purchased in bulk from a scrap dealer. This also reflects upon the valuation methods of core inventories.

The goal, then, for the first tier of the website was modified simply to provide contact information and a look-up resource for parts numbers.

Matching the computer infrastructure for an E-Commerce website was also deemed problematic after careful consideration. Mr. Van Wechel had become acutely aware of the vulnerability of his network and vital data system by having his primary server connected to the Internet. The time spent configuring and assessing logs in Sygate Firewall revealed a nearly constant barrage of scans and access attempts that were blocked by the system. This was corroborated by the log of the router. The graduate student also indicated that these two firewalls

were effective against casual or roaming attacks, but are probably inadequate against a targeted attack.

DPI presently receives email through accounts established through Verizon, the DSL provider. Because of this, no plan was made to add an email server system. But it became evident that if E-customers were going to submit even basic web forms requesting parts or sales information, they would have to do so with their own email accounts, because the web server would need an outgoing mail server to send the request to DPI's email.

So, the prospect of any real Ecommerce implementation dwindled, but the site design continued anyway.

The site was divided into four basic pages; A Home or welcome page, a products page to describe general matters about calipers, a product lookup page, and an About-DPI page, to offer some historical information and a connection to the real people in the company. A view of each main page is given in figures 9, 10, 11, and 12.

The product lookup page links to several sub pages with caliper information for the general consumer, and to a browser based interface to the MySQL database designed for administrative use. Examples of these sub pages are given in figures 13-17.



Figure 9: Screen Capture - DPI Website Index.php

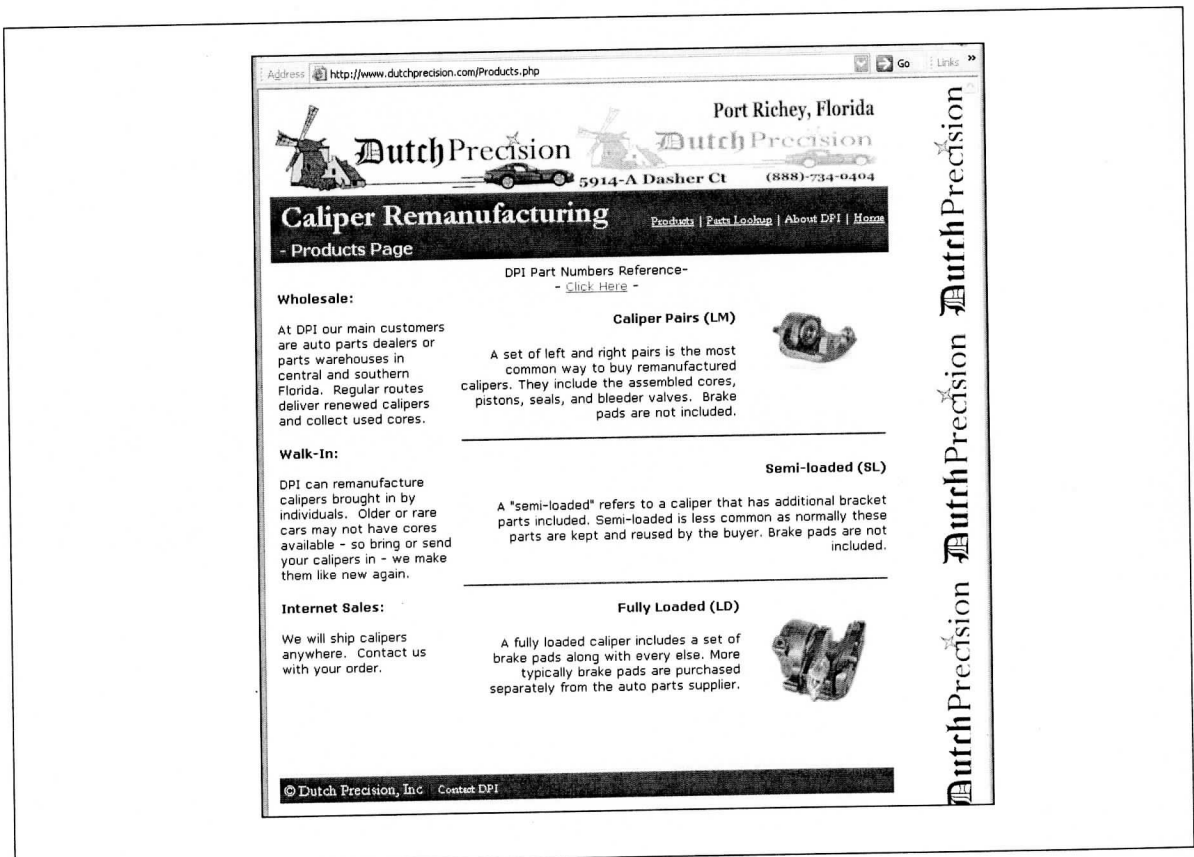


Figure 10: Screen Capture - DPI Website Products.php

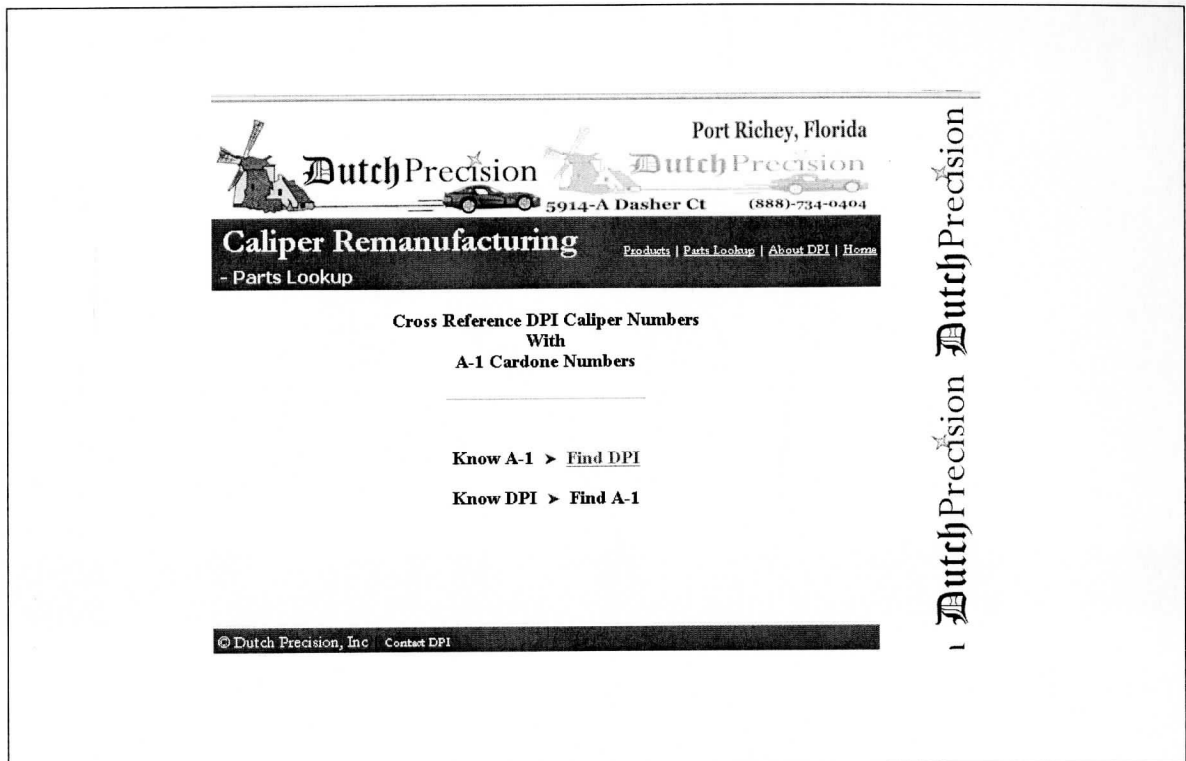



Figure 11: Screen capture - DPI Website PartsLookup.php



Figure 12: Screen Capture - DPI Website About DPI.php page


Port Richey, Florida

5914-A Dasher Ct (888)-734-0494

Caliper Remanufacturing [Products](#) | [Parts Lookup](#) | [About DPI](#) | [Home](#)

- Parts Lookup

A1 Cardone to DPI

- Jump to Section -












[1](#) | [2](#) | [3](#) | [4](#) | [5](#) | [6](#) | [7](#) | [8](#) | [9](#) | [10](#)

A-1	DPI	Side	DESCRIPTION	YEAR
16-4002	LM129	R	CH CHEVETTE 10rim	78-79
16-4003	LM130	L	CH CHEVETTE 10rim	78-79
16-4004	LM127	R	SKYHAWK, SUNBIRD	79-81
16-4005	LM128	L	SKYHAWK, SUNBIRD	79-81
16-4006	LM113	R	GM FULL SIZE	77-78
16-4007	LM114	L	GM FULL SIZE	77-78
16-4008	LM109	R	FO FAIRMONT, ME ZEPHYR	78
16-4009	LM110	L	FO FAIRMONT, ME ZEPHYR	78
16-4010	LM010	R	FO LTD, T-BIRD, CONTINENTAL	68-72
16-4011	LM011	L	FO LTD, T-BIRD, CONTINENTAL	68-72
16-4012	LM012	R	FO MUSTANG, COMET, COUG	68-73
16-4013	LM013	L	FO MUSTANG, COMET, COUG	68-73
16-4014	LM117	R	AM JEEP CJ5,6,7	77-81
16-4015	LM118	L	AM JEEP CJ5,6,7	77-81

Figure 13: Screen Capture - Sub Parts Lookup – html lists

Browser Based Access to MySQL Tables

DPI Database Administration

-  **Make**
-  **ModelTbl**
-  **CaliperTbl**
-  **CalipModelJct**
-  **YearTbl**
-  **SubMoTbl**
-  **CardoneRefTbl**
-  **USBrakeRefTbl**
-  **ApraCalRefTbl**
-  **DpCompTbl**
-  **VehicleTbl**

Powered by [AppGini 2.61](#)

Figure 14: Screen Capture – Browser based database Admin

Make

Make_ID	Make_Name	Abbv
1	AC COBRA - ROADSTER	ACC
2	ACURA	ACU
3	ACURA TRUCKS	ACT
4	ALFA ROMEO	ALF
5	AM GENERAL	AMG
6	AMC	AMC
7	ASTON MARTIN	AST
8	AUDI	AUD
9	AUSTIN	AUS
10	AVANTI - STUDEBAKER, HAWK	AVA

Records 1 to 10 of 95

Detail View

Make_ID 7

* Make_Name

Abbv

Figure 15: Screen capture – Browser based data entry

Filters

	Filtered field	Comparison Operator	Comparison Value
Filter01	Make_ID	Like	%Au%
Filter02	And		
Filter03	And		
Filter04	And		
Or			
Filter05	Model_ID	Like	%fo%
Filter06	And	Equal to Not equal to Greater than Greater than or equal to Less than Less than or equal to Like Not like	
Filter07	And		
Filter08	And		
Filter09			
Filter10	And		
Filter11	And		
Filter12	And		

Figure 16: Screen capture - Database Query Filters

VehicleTbl

PRINT PREVIEW FILTERS RESET FILTERS Go to table

V_ID	Make_ID	Model_ID	Year_ID	Sub_ID
72	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	68-71	7mm expander bar
73	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	90-94	No expander bar
74	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	72-89	No expander bar
75	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	68-71	No expander bar
76	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	90-94	REAR
77	ALFA ROMEO	2000 - Spyder Convertible, GTV Coupe, Sedan, Graduate, Quadrifoallo	72-89	REAR
111	AUDI	A4, A4 QUATTRO	00-03	REAR EXC QUATTRO
112	AUDI	A4, A4 QUATTRO	98-99	REAR EXC QUATTRO
113	AUDI	A4, A4 QUATTRO	96-97	REAR EXC QUATTRO
114	AUDI	A4, A4 QUATTRO	95-01	REAR QUATTRO

Records 1 to 10 of 154

Detail View

* V_ID ADD NEW

* Make_ID

* Model_ID

* Year_ID

* Sub_ID

Local intranet

Figure 17: Screen capture - Sample Query Result

Further discussion of the database related web pages will be address in the next section.

Software: Continuing, then, the four main pages of the DPI website are coded in PHP and HTML, with some use of cascading style sheets (CSS), and one small segment of JavaScript, used to cycle through a list of four images on the home page. MS FrontPage 2003 was used whenever possible to generate html code, though most of the PHP was done using a plaintext editor, Windows Notepad. The JavaScript was found on the Internet and adapted to work on this site. The source of this code is credited in the code itself.

The PHP was used primarily to call or include pages of reusable coding. In particular, each page includes a header, navigational menu, individual page identifiers, and a common footer file. A listing of these files is included in Appendix B along with much of the code used. One point of interest to the graduate student was the inability of MS FrontPage to manage links found in PHP included files. The main pages were located on in the root directory with the included headers and such located in a sub folder. Any links residing inside the included files, such as to images, are referenced based on the root location of the calling file, not relative to the location of the include. FrontPage would create anchors relative to the includes. While this was not difficult to repair, it negated much of the convenience of this WYSIWYG editor.

Imaging: The final element of the web design was the creation of suitable graphics for the site. All graphics were created or manipulated using a digital camera and MS PhotoDraw. Two examples are included in figures 18 and 19 following this discussion, and others are viewable in Appendix D.

The Dutch Precision logo is used on business cards, stationary, and other similar places. Several new logo variations were created to provide fresh options for the website. Automobiles and calipers are obviously related to DPI, as are any number of icons associated with the Dutch, Holland, or the

Netherlands. Windmills, wooden shoes, Dutch architecture, tulips, and delft colorings were images considered in one way or another.

Ways to expand upon the word 'Precision' were also explored. A Dutch windmill tends to convey a sense of tradition and connection to the past, but the word 'Precision' is more likely associated with industry or accuracy – or with cutting edge technology. These ideas were already combined in the existing logo with a windmill, ornate Old English font for lettering, and the leading edge of an automobile. Unfortunately, the word 'Dutch' was too easily hidden in the windmill, and it was difficult to recreate the image for the website in a pleasing manner. As with most image or advertising matters, there is still discussion at DPI about the best logo design, but one can see from the previous web pages screen shots what was adopted for the site.

The new logo is intended to represent the stability and trustworthiness of old world values, but applied with 'sports car' technology. The lettering font is different for each word, also intended to convey these values.

The graduate student has very little training in digital graphics, and one of the difficulties encountered was the variation in coloration between the laptop computer display used to create the images, and those used at DPI. Many variations were advanced to account for these differences.

Most of the images created were not ultimately used in the website, but they still reflect significant investment of time and creative energy.



Figure 18: DPI Business Card



Figure 19: Recreated Logo

Database Development:

Without question, the most difficult problems encountered in this project were database related. The graduate student underestimated the difficulties and frustrations of this aspect of the project. While fundamentally grounded in database design, this was not the student's area of expertise.

That being said, the original goal was to provide customer access to Dutch Precision's caliper data stored in a MySQL database. This could lead to allowing customers to access their individual account information or the status of their orders. In addition, the system should strive to be user friendly, with data updates enacted with reasonable ease by Mr. Van Wechel, or someone with similar computer skills.

Data Alignment: Reconstructing DPI spreadsheet data into well formed database tables was the single most time consuming factor in this project. These

spreadsheets were created by Mr. Van Wechel from the printed parts books of other caliper companies in the industry – in particular, A-1 Cardone, U.S. Brake, and Apra Automotive. Mr. Van Wechel kept vehicle specific caliper data in one spreadsheet and most of his pricing and description data in another. While the workbooks referred to the same calipers and parts numbers, they were often entered manually, which made them prone to errors, abbreviations, and data mismatches.

Figure 20 below provides an example of the way the data was organized.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T		
1	mount/# of Pistons* = check mount position										DPI	DPI		PAD #	CASTING #		CARDONE		US BRAKE		APRA	
2	MAKE / MODEL / DESCRIPTION										LEFT	RIGHT	LOADED SET	LEFT	RIGHT	LEFT	RIGHT	LEFT	RIGHT	LEFT	RIGHT	
3	AC COBRA - ROADSTER																					
4						GIR 3-PIS. VR														30072	30071	
5						GIR 3-PIS. NVR														31342	31341	
6						REAR 3-PIS.					D9									30068	30067	
7						REAR 2-PIS.	7582	7581			D9	64327787/8	7791/2	769	768	7340	7339	31255	31256			
8	ACURA																					
9	CL SERIES																					
10		1	01-03					8320	8319		D787			2584	2585					32857	32858	
11		1	97-99	6	CYL			7972	7971		MD503	17CL15VN		1461	1460	7366	7365	30025	30026			
12		1	97-99	4	CYL			7954	7953		MD465	2224, 2227		1335	1334	6025	6024	31111	31112			
13		1	01-03	REAR				8100	8099		D537			2589	2588					32859	32860	
14		1	97-99	REAR				8102	8101		MD536, 7	3402		1823	1822	7572	7571	30163	30164			
15	INTEGRA																					
16		1	00-01	Exc. type R				8098	8097		MD617	17CL14VN (T7)		1734	1735	7380	7379	31071	31072			
17		1	00-01	type R				7972	7971		MD503	17CL15VN		1461	1460	7366	7365	30025	30026			
18		1	94-99	w/o pad anti-rattle clip				8098	8097		MD617	17CL14VN (T7)		1734	1735	7380	7379	31071	31072			
19		1	90-93	w/ pad anti-rattle clip				7712	7711		MD341	17CL14VN		1005	1004	3989	3988	31075	31076			
20		1	86-89					7382	7381		MD334	2145		1003	1002	3690	3689	31067	31068			
21		1	00-01	REAR Type R							D537											
22		1	94-01	REAR exc. above				8030	8029		MD339	7CLP13S (R3)		1557	1556	7442	7441	31073	31074			
23		1	92-93	REAR				8018	8017		MD374	7CLP13S NCS		1402	1401	3995	3994	31069	31070			
24		1	90-91	REAR				8018	8017		MD374	7CLP13S NCS		1402	1401	3995	3994	31069	31070			
25		1	86-89	REAR				7692	7691		MD339	7CLP13S		967	966	3987	3986	30007	30008			
26	LEGEND 2-DOOR COUPE																					
27		2	94-96					7416	7415		MD503	17CL15VN		1745	1744	7382	7381	30011	30012			
28		1	91-93					8318	8317		MD503	17CL15VN 23T		1463	1462	7265	7264	30015	30016			
29		1	87-90					7712	7711		MD341	17CL14VN		1005	1004	3989	3988	31075	31076			
30		1	91-96	REAR				7798	7797		MD536	10CL15SN		1449	1448	7127	7126	30013	30014			
31		1	87-90	REAR				7984	7983		MD365	11CLP13S (GO)		965	964	3991	3990	30017	30018			
32	LEGEND 4-DOOR SEDAN																					
33		2	94-96	GS Series				7416	7415		MD503	17CL15VN		1745	1744	7382	7381	30011	30012			
34		1	91-95	Exc. GS Series				8318	8317		MD503	17CL15VN 23T		1463	1462	7265	7264	30015	30016			
35		1	86-90					7712	7711		MD341	17CL14VN		1005	1004	3989	3988	31075	31076			

Figure 20: Sample view of DPI data spreadsheet

The first column holds Manufacturer or Make data, and the next column is used for the models. A single make entry corresponds to a sequence of models,

and a range of years, though the actual make entry resides on a row or line of its own. Columns may be blank to provide visual spacing, or they may contain notes and comments regarding specific installations. And, while generally consistent, there are numerous differences between data within a single column.

Date data, for example, may reflect a single year, 89, or 1989, or with a hyphen 89- representing the start of a series of years. Most commonly dates reflect a start and stop date range for a given model, as in 98-02. Less often, but still numerous, dates include mid-year production starts or stops, such as 7/86-88. Date data was most often in text or character format, less often numerical, and almost never in date format.

In general, a knowledgeable reader could decipher short hand notes, the association of columns and rows, or the reasons for diverse markers of missing data. Yet with over 7000 records, this proved to be a significantly daunting task.

Generally, the sorting, filtering, and lookup features of Excel were used to fill in missing data, refine data to unique instances, trim leading and trailing spaces, and enforce consistent data types within columns. Formulas that came most in handy were Trim(), T(), Match, VLookup, Text(), and a variety of filters for ordering data, creating unique data sets, and eliminating spaces.

The goal was to generate a simple relationship between DPI caliper numbers and A-1 caliper numbers, along with a description of the make, model, and year of application. The task seemed to be attainable and reasonable.

Online research of other auto parts websites revealed a variety of means for 'drilling' down to identify make, model, and year of an automobile. But always it included well defined date data – and this was nearly impossible to extract from the DPI spread sheets. The student become aware of start date and end date means of identifying date ranges, but frankly the diversity of date data made it difficult to apply such methods with consistency.

Regardless, the student was never able to completely reconcile all of the data, and less than optimal database normalization and data types resulted to allow data to remain as text - for dates, for example.

Database Design: The database table design adopted is as follows, and figure 21 shows a late iteration of data joins and cardinalities:

Database: DpiCalipersDb

MakeTbl (Make ID, Make_Name, Make_Abbv)

ModelTbl (Model ID, Model_Name, Make ID, YrRange ID, Sub ID)

YearTbl (Range ID, YrRange)

SubMoTbl (Sub ID, SubValue)

CalipModelJct (Model ID, DpCal ID)

CaliperTbl (DpCal ID, Dpi_Num, Side, Cal_Descr, Casting)

CardoneRefTbl (A1 ID, A1_Num, DpCal ID)

USBrakeRefTbl (USB_ID, USB_Num, DpCal_ID)

ApraRefTbl (Apra_ID, Apra_Num, DpCal_ID)

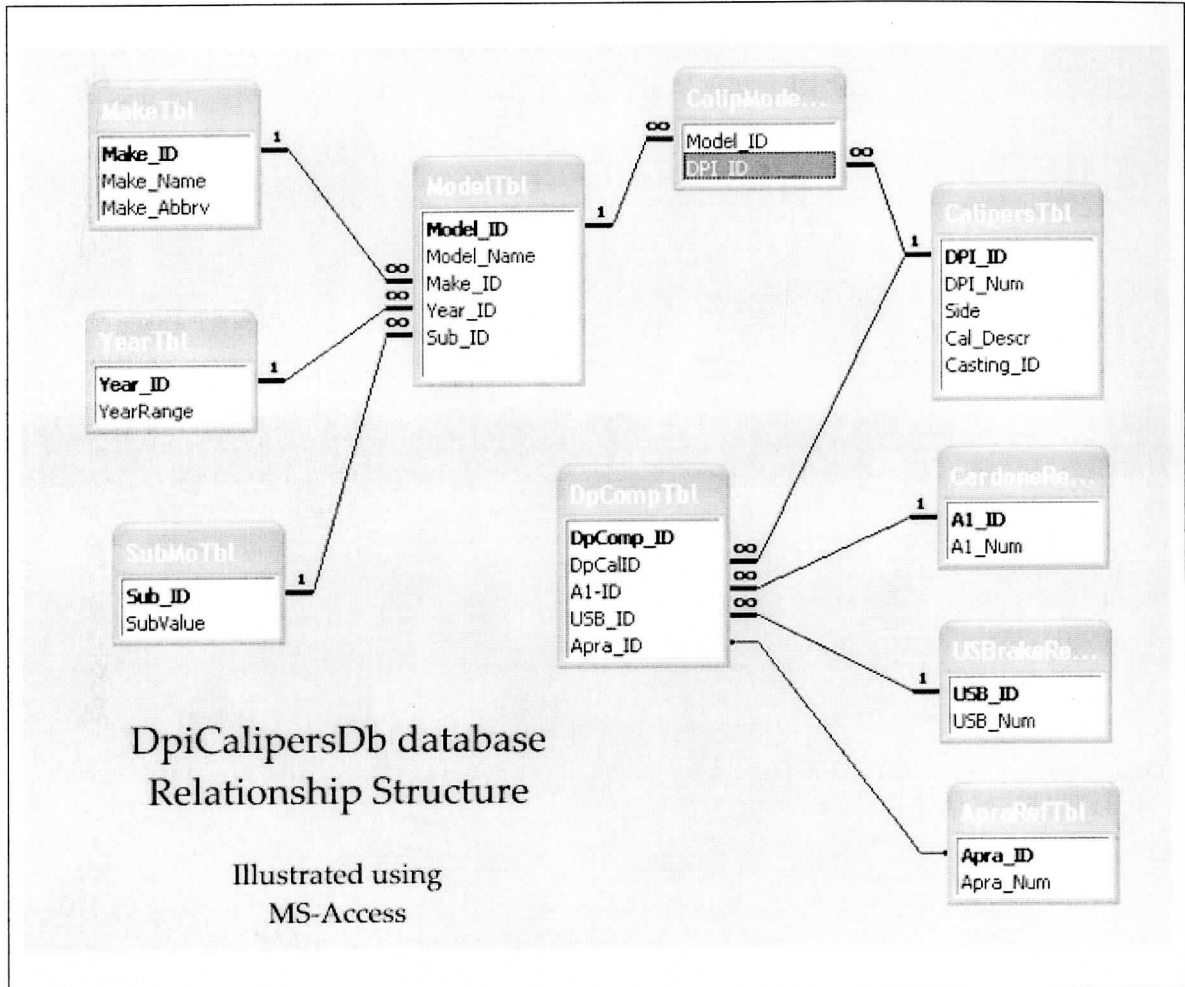


Figure 21: Design Structure and Cardinalities of Web based caliper reference database

The open source nature of MySQL database makes it a very attractive option for small businesses, but additional responsibility falls to the user to learn and utilize the online documentation. As stated earlier, one goal was for Mr. Van Wechel to understand the workings of his website. To that end, a software application named AppGini Professional (v. 2.61) was purchased to help create an online interface to the database. AppGini was helpful in generating the code

for the MySQL database, once it was given the design specifications, and then making the Php code for a browser interface with the data.

The results of AppGini were to create a satisfactory administrative web portal to the database, and these results are illustrated previously in figures 14-17. However, significant security concerns are raised when administrative access is gained via the same web server used by the general public. Passwords and other user authorization are required. Folder permissions can be granted through Windows, and through Apache server htaccess and configuration files.

By this time, Dutch Precision has decided not to further implement the web database on the primary DPI server, but rather to arrange an E-commerce site with a reputable Internet Service provider. But until that step is taken, html-list pages are accessible to visitors at the Dutch Precision website to lookup A-1 and corresponding DPI caliper numbers.

Before summarizing this project and drawing final conclusions, the final section addresses areas of relevant research, including what was learned about government contracts.

Findings and Discussions

As first expressed in the project proposal, the decision to use the open-source software applications, MySQL, Php, and Apache 2, reflects growing trends in small to mid-sized businesses.

Apache: According to the December 2004 survey (figures 22 and 23) by Netcraft.com, the well regarded Internet utility observer from United Kingdom, there are now over 58 million web domains hosted on the Internet, 27 million classified as “active”, and approximately 68% are on Apache servers, up nearly 3% since May (Netcraft.com, 2004). The nearest competitor, Microsoft IIS, is holding steady with 21% as other servers are falling away. As mentioned in pre-project research, Microsoft IIS still dominates server usage amongst Fortune 1000 companies (Serverwatch.com, 2003).

This growth in Apache comes after the release of Apache 2, which adds multi-thread capabilities for Windows-based implementation, which the widely used and stable Apache 1.3 did not provide. In spite of the fact that Apache 2 has required a series of security patches, it has seen more than double comparative growth on Windows operating systems since 2003.

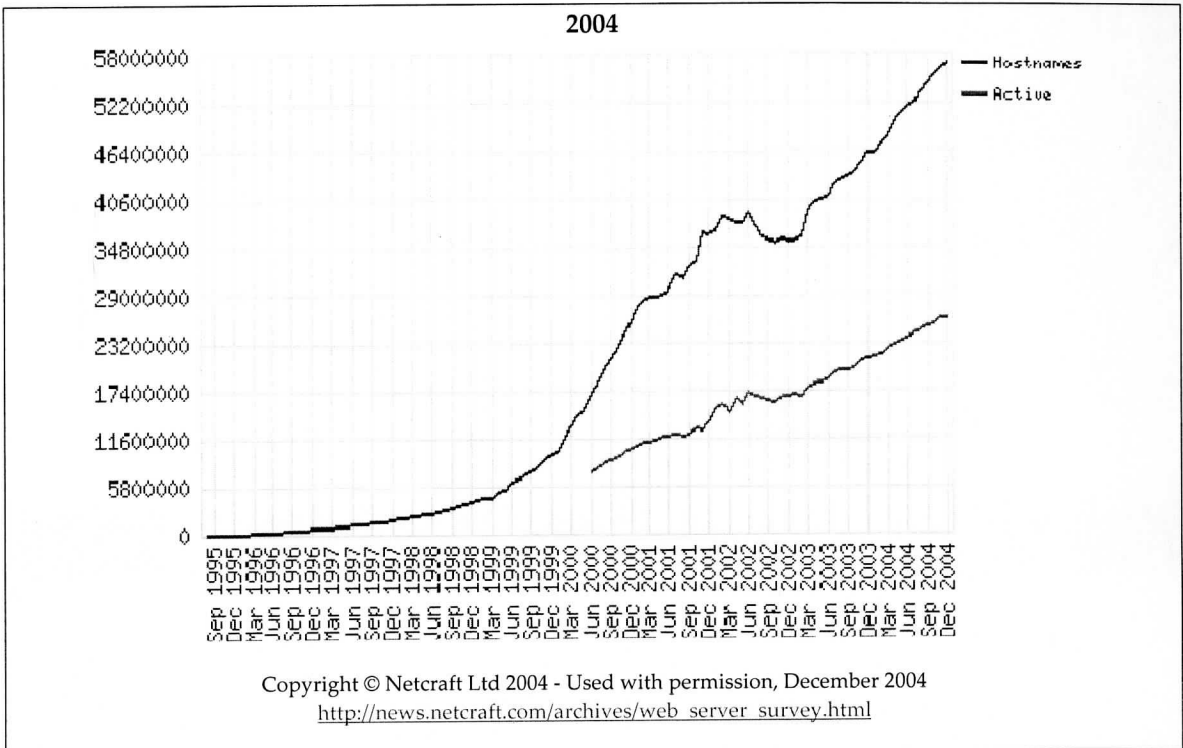


Figure 22: Number of Hostnames and Active Internet Sites

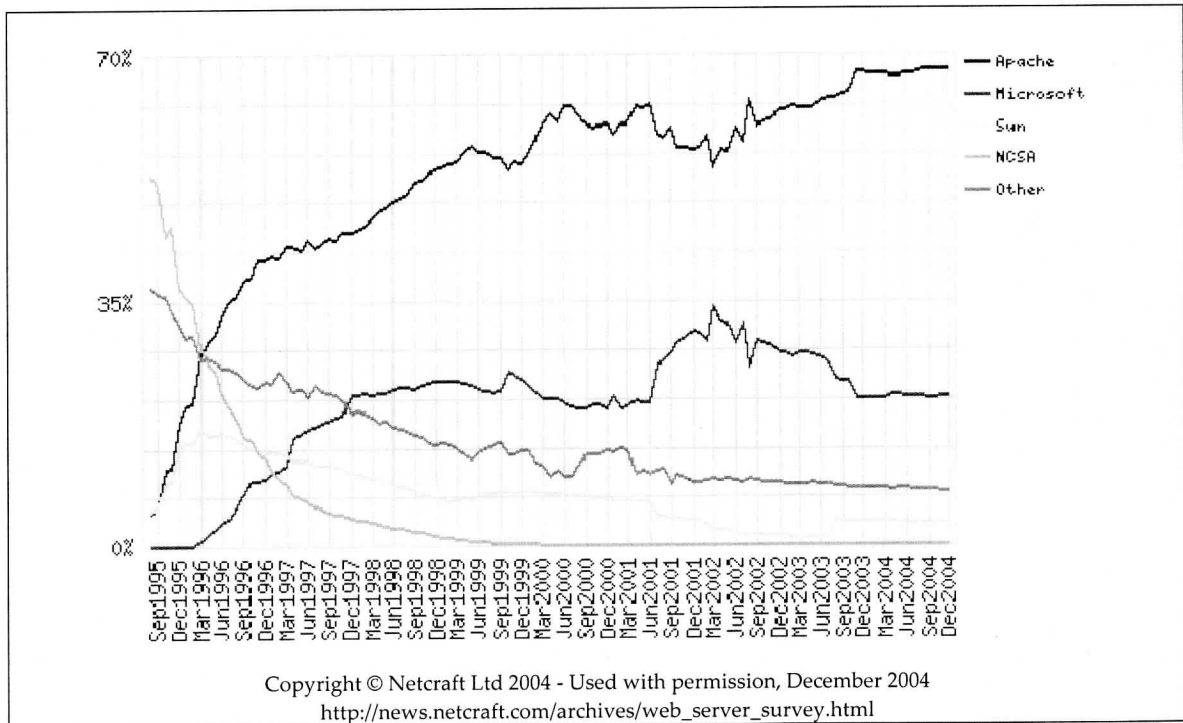


Figure 23: Market Share for Top Servers Across All Domains August 1995 - December 2004

PHP: Although PHP is widely thought of as implying Linux, Apache and MySQL, nearly 7% of PHP sites (when counting by IP address) run on Windows. This has doubled over the last year (2003), and on its current growth trajectory PHP will overtake Cold Fusion as the most popular non-Microsoft scripting language used on Windows during the next year (see figure 24).

Also, as mentioned in the initial proposal, Microsoft's Active Server Pages (ASP) has gone through a rather radical shift changing from VBscript in its IIS.4 to using C# in Asp.net IIS.6, caused considerable adjustments for developers.

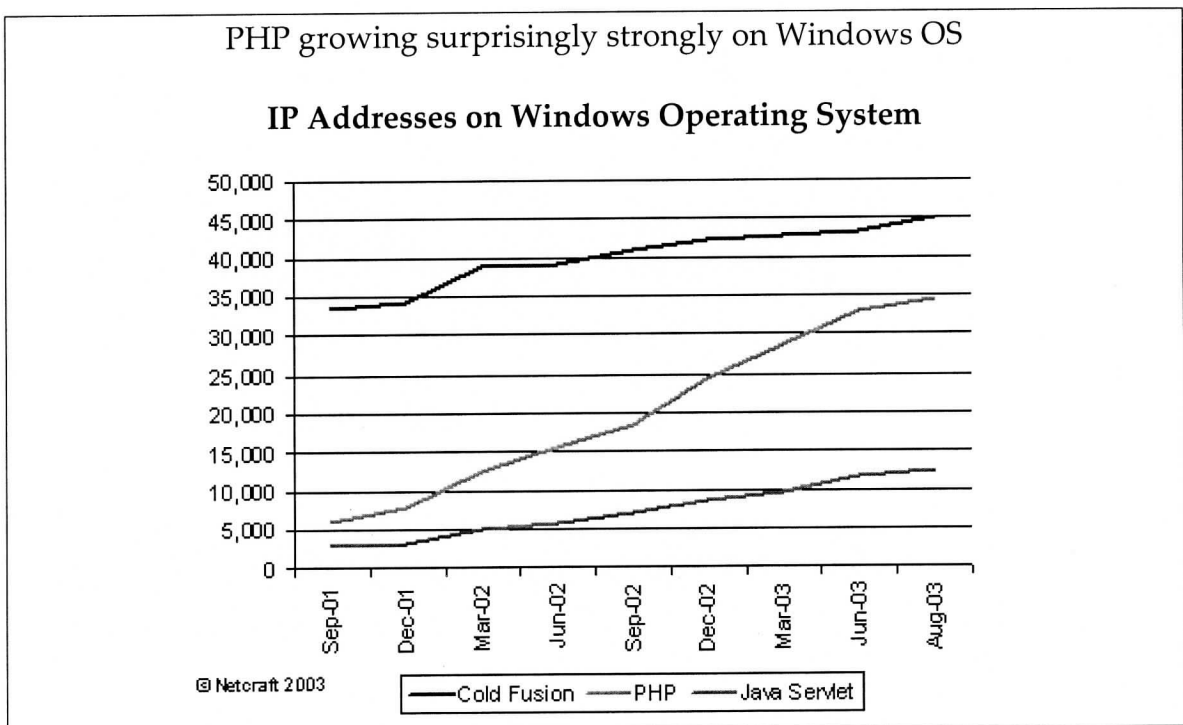


Figure 24: PHP Growth

MySQL: MySQL still provides extensive functionality and competitive speed for moderate usage web sites, but it does so at substantially lower overall cost.

Dutch Precision is most likely representative of the mind set of many small

businesses when it comes to making technology decisions in favor of lower costs. Whether wisely or not, DPI will often try to “make do” with the technology in hand or that is least expensive rather than pursuing more powerful or more costly options. A December 2004 online cost comparison listed MS SQL (5 user basic service) for around \$1,500, and Oracle (8i – 5 users) right at \$5,000. This helps make the \$0.00 initial cost MySQL an even more attractive option for the small business owner.

All in all, Web site developers appear to be less dogmatic in their application choices than many analysts like to portray, and are prepared to mix development paradigms when it suits their purpose. Scripting languages are seeing the best growth of any class of web technologies presently as site owners seek to make their sites more sophisticated and see a good return on the development time invested (Netcraft.com, 2004).

Government Contracts

The final problem or goal identified by the original proposal was to pursue compliance with government standards for online company disclosure to help qualify for bidding on government contracts. Much of the following information was gathered from the Florida based, Assistant Director of Small Businesses, Ms.

Sandra J. Scanlan. She serves the Defense Contract Management Agency in St. Petersburg.

Ms. Scanlan notes, in training sessions, that Government contracting is identified as the biggest business in the world, with the United States Government awarding a new contract every 20 seconds of every working day worth, on average, \$465,000. Government agencies involved in purchasing are numerous and varied, but in general each is looking for quality products and services, with dependable delivery schedules at fair and reasonable prices. Timely delivery of products is often of greater consideration than overall costs.

For a small business to get started in government contracting, they must identify their code status in these areas:

- Standard Industrial Classification Codes (SIC)
- North American Industrial Classification Codes (NAICS)
- Federal Supply Class Codes (FSC)
- Acquire a DUNs Number

The next step is to register the business with Central Contractor Registration (CCR – www.ccr.gov). This is required to do any business with the Department of Defense. The business must also register with Pro-Net (www.sbs.gov) which is a database of small businesses. This database is then searched by Government buying agencies.

For Dutch Precision, located in Florida, some sources of assistance are the regional Procurement Technical Assistance Centers (PTACs), located in Pensacola, Tampa, Ft Myers, Jacksonville, and Miami. Also the Small Business Development Center, the Contract Management Agency in Orlando and Clearwater, and the Service Corps of Retired Executives (SCORE.org), also located in Clearwater.

Conclusions and Future Work

The success of the project, "E-Commerce Implementation for Dutch Precision, Inc. ", can be assessed by diverse measures. When viewed from a strict business perspective - one that evaluates the costs, returns, and structured timelines of a proposed plan - this project has decidedly mixed results. Dutch Precision realized direct network improvements from the graduate student, and has a web presence available to the world. Dutch Precision also became acutely aware of the difficulties and dangers involved in hosting ones own E-commerce web site. A tangible measure was achieved in understanding the value of an E-commerce ready website.

Less success was realized, however, in the depth of information made available on the current web site. Also, the process extended some three months beyond the expected schedule, or nearly twice as long. Had there been budgets involved, the time and cost over-run would be significant.

From an educational point of view, the project was a resounding success, and served as a capstone training mechanism for nearly every aspect of the MSIS degree. The student encountered and learned from a wide variety of networking, software, database, research, and project management issues. He was stretched in each of these areas and brings a new sense of realism to the understanding and application the MSIS degree.

References

Apache 2.05, Http Server: Software - <http://httpd.apache.org/>

AppGini 2.6, PHP and MySQL script generation: Software - <http://www.bigprof.com/appgini/>

Defense Contract Management Agency (DCMA), [Online] (October, 2004)
<http://www.dcma.mil/>

InformationWeek, (Nov 6, 2003) pNA "Open-Source Scripting Language Becoming Dominant; Netcraft says PHP is found on 52% of the 14.5 million Apache-based Web sites that it inspected."

Krill, P. (December 12, 2003). InfoWorld Media Group, Inc.. "Open source said on cusp of broad acceptance; Apache founder anticipates vast expansion."

MySQL 4.1, Database Server Software: Software -
<http://dev.mysql.com/downloads/mysql/4.1.html>

Netcraft.com, 2004. [Online] Referenced on May 1, 2004
http://news.netcraft.com/archives/web_server_survey.html

Orgill, E. (January 2004). Evans Data Corporation, "Open Source Database Development Closes In On Microsoft."

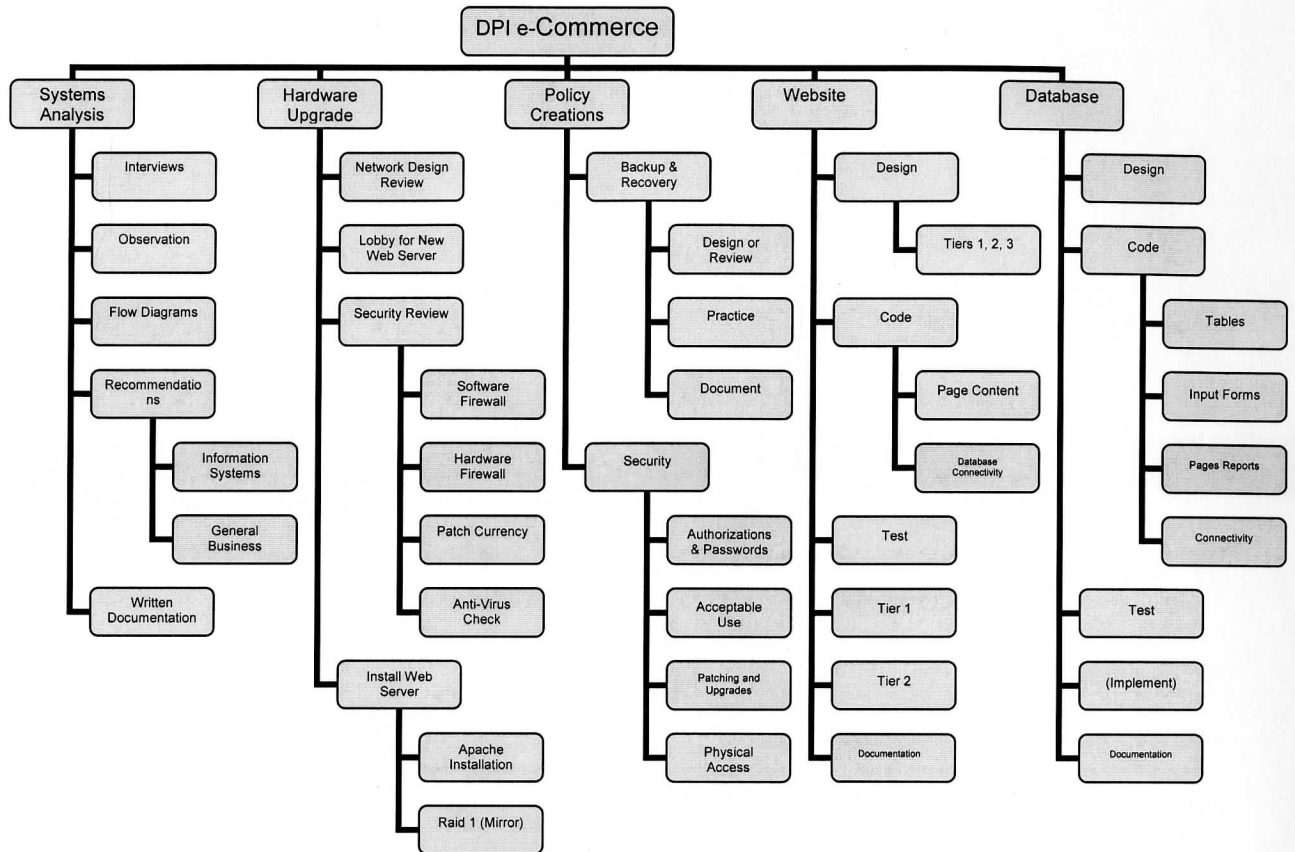
PHP 4.39, Server-side scripting language: Software - <http://www.php.net/>

Roundtable Software , Armor Accounting Systems: Software -
<http://www.cenflacc.com/mainmenu.htm>

Scanlan, Sandra J. (2003). "Introduction to Government Contracting," PowerPoint Resources, St. Petersburg, Florida.

Serverwatch (2004). [Online] <http://www.serverwatch.com>

Appendix A: Work Breakdown Structure and Gantt Project Task Schedule



Gantt Project Event Schedule

ID	Task Name	Duration	Start	Finish
1	TravelOut	1 day	Sun 5/9/04	Sun 5/9/04
2	Systems Analysis	8 days	Sun 5/9/04	Mon 5/17/04
3	Observation	2 days	Sun 5/9/04	Mon 5/10/04
4	Interviews	2 days	Sun 5/9/04	Mon 5/10/04
5	Flow Diagrams	2 days	Tue 5/11/04	Wed 5/12/04
6	Recommendations	1 day	Thu 5/13/04	Thu 5/13/04
7	Info Systems	1 day	Thu 5/13/04	Thu 5/13/04
8	General Business	1 day	Thu 5/13/04	Thu 5/13/04
9	SA Documentation	2 days	Sat 5/15/04	Mon 5/17/04
10	Hardware Upgrade	21 days	Thu 5/13/04	Thu 6/17/04
11	Network Design/Review	1 day	Thu 5/13/04	Thu 5/13/04
12	Lobby for New Server	1 day	Thu 5/13/04	Thu 5/13/04
13	Security Review	1 day	Wed 6/16/04	Wed 6/16/04
14	Firewall - Software	1 day	Wed 6/16/04	Wed 6/16/04
15	Firewall - Hardware	1 day	Wed 6/16/04	Wed 6/16/04
16	Patch Currency	1 day	Wed 6/16/04	Wed 6/16/04
17	Anti-Virus Check	1 day	Wed 6/16/04	Wed 6/16/04
18	Install Web Server	1 day	Thu 6/17/04	Thu 6/17/04
19	Apache Install	1 day	Thu 6/17/04	Thu 6/17/04
20	Raid 1 Config	1 day	Thu 6/17/04	Thu 6/17/04
21	Policy Creation/Review	7 days	Mon 6/7/04	Wed 6/16/04
22	Backup/Recovery	6 days	Mon 6/7/04	Tue 6/15/04
23	Design/Review	2 days	Mon 6/7/04	Tue 6/8/04
24	Practice	2 days	Wed 6/9/04	Thu 6/10/04
25	Document	2 days	Mon 6/14/04	Tue 6/15/04
26	Security	1 day	Wed 6/16/04	Wed 6/16/04
27	Authorizations/PWs	1 day	Wed 6/16/04	Wed 6/16/04
28	Acceptable Use	1 day	Wed 6/16/04	Wed 6/16/04
29	Patching/Upgrades	1 day	Wed 6/16/04	Wed 6/16/04
30	Physical Access	1 day	Wed 6/16/04	Wed 6/16/04
31	Website	15 days	Mon 5/24/04	Tue 6/15/04
32	Design	5 days	Mon 5/24/04	Fri 5/28/04
33	Tier1	1 day	Mon 5/24/04	Mon 5/24/04
34	Tier2	2 days	Tue 5/25/04	Wed 5/26/04
35	Tier3	2 days	Thu 5/27/04	Fri 5/28/04
36	Code	4 days	Mon 5/31/04	Fri 6/4/04
37	Page Content	4 days	Mon 5/31/04	Fri 6/4/04
38	Database Connectivity	4 days	Mon 5/31/04	Fri 6/4/04
39	Test	2 days	Mon 6/7/04	Tue 6/8/04
40	Implement Tier 1	2 days	Wed 6/9/04	Thu 6/10/04
41	Documentation	2 days	Mon 6/14/04	Tue 6/15/04
42	Database	63 days	Wed 6/16/04	Tue 9/14/04
43	Design	3 days	Wed 6/16/04	Mon 6/21/04
44	Code	16 days	Tue 6/22/04	Wed 7/14/04
45	Tables	4 days	Tue 6/22/04	Mon 6/28/04
46	Input Forms	4 days	Tue 6/29/04	Fri 7/2/04
47	Output Pages/Reports	4 days	Mon 7/5/04	Thu 7/8/04
48	Web Connectivity	4 days	Fri 7/9/04	Wed 7/14/04

49	Test	4 days	Thu 7/15/04	Tue 7/20/04
50	Document	4 days	Wed 7/21/04	Mon 7/26/04
51	Write Final Paper	10 days	Tue 7/27/04	Mon 8/9/04
52	Present		Tue 9/14/04	Tue 9/14/04


```

Telephone<br>
  </b></u>Local 727-846-9474<br>
  Toll Free 888-734-0404<br>
  Fax 727-846-0791</font></p>
<p align="center" style="text-align: left"><font face="Verdana">Email:<br>
  <a href="mailto:dutch.precision@verizon.net?subject=Message from DPI
  Website" style="color: white">
<font color="#816500">dutch.precision@verizon.net</font></a></font></p>

&nbsp;</td>
</tr>
</table>
</div>

<?php
  require('Includes/DpiFooter.htm');
?>

<!-- This is the Products.php page for the DPI Website -->
<?php
  require('Includes/DpiHeader.htm');
  require('Includes/DpiMenu2.htm');
  require('Includes/DpiMenuAddProd.htm');
  // require('Includes/Products.htm');
?>

<!-- Content of Products Page -->
<div align="left">

  <table border="0" cellpadding="1" cellspacing="0" id="table4" width="650">

    <tr>
      <td>

<table border="0" cellpadding="5" style="border-collapse: collapse"
width="100%" id="table6">
  <tr>
    <td width="182" rowspan="3">
      <p align="left" style="text-align: left">
        <font size="2" face="Verdana"><b>Wholesale:<br>
        </b><br>
        At
        DPI our main customers are auto parts dealers or parts warehouses in
        central
        and southern Florida.&nbsp;  Regular routes deliver renewed calipers
        and collect used cores.</font><p align="left" style="text-align: left">
        <font size="2" face="Verdana"><b>Walk-In:<br>
        </b><br>
        DPI can remanufacture calipers brought in by individuals.&nbsp;  Older or
        rare cars may not have cores available - so bring or send your calipers in
        -
        we make them like new again.</font><p align="left" style="text-align:
        left">
        <font size="2" face="Verdana"><b>Internet Sales:<br>
        </b><br>
        We will ship calipers anywhere.&nbsp;  Contact us with your
        order.</font><p>
        <font size="3" face="Garamond">&nbsp;  </font></td>
      <td width="339">
        <p align="left" style="text-align: center"><font size="2"
        face="Verdana">DPI Part Numbers Reference-
        <br>
        - <a href="PartsLookup.php" id="OnWhiteLink">Click Here</a> - </font></p>

```

```

        <p style="text-align: right"><b><font size="2" face="Verdana">Caliper
Pairs
(LM)</font></b></p>
        <p style="text-align: right"><font size="2" face="Verdana">&nbsp;A set of
left and right pairs is the most common way to buy remanufactured
calipers.
        They include the assembled cores, pistons, seals, and bleeder
valves.&nbsp;
        Brake pads are not included. </font></td>
        <td>
        <p align="center" style="text-align: center">
        </td>
        </tr>
        <tr>
        <td colspan="2"><hr color="#5A3C03">
        <p align="right" style="text-align: right"><font size="2"
face="Verdana"><b>
Semi-loaded (SL)</b></font></p>
        <p align="left" style="text-align: right"><font size="2"
face="Verdana">&nbsp;A
'quote;semi-loaded'quote; refers to a caliper that has additional bracket
parts
        included. Semi-loaded is less common as normally these parts are kept and
reused by the buyer. Brake pads are not included.</font></p>
        <hr color="#5A3C03"></td>
        </tr>
        <tr>
        <td width="339">
        <p align="right" style="text-align: right"><font size="2"
face="Verdana"><b>
Fully Loaded (LD)</b></font></p>
        <p align="left" style="text-align: right"><font size="2"
face="Verdana">&nbsp;A
fully loaded caliper includes a set of brake pads along with every else.
More typically brake pads are purchased separately from the auto parts
supplier. </font></p>
        <p>&nbsp;</td>
        <td>
        <p align="center" style="text-align: center">
        </td>
        </tr>
</table>
<p>&nbsp;</td>
</tr>
</table>
</div>

<?php
require('Includes/DpiFooter.htm');
?>

<!-- This is the PartsLookup.php page for the DPI Website -->
<?php
require('Includes/DpiHeader.htm');
require('Includes/DpiMenu2.htm');
require('Includes/DpiMenuAddPartsLk.htm');
// require('Includes/PartsLookup.htm');
?>

<!-- Content of LookUp Page -->
<div align="left">

```

```

<table border="0" cellpadding="1" cellspacing="0" id="table4" width="650">

<tr>
  <td align= "center">
    <br>
    <H2>Searchable parts database coming soon!</h2>
  </td>
</tr>
</table>

<p>&nbsp;</td>
</tr>
</table>
</div>

<?php
  require('Includes/DpiFooter.htm');
?>

<!-- This is the AboutDPI.php page for the DPI Website -->
<?php
  require('Includes/DpiHeader.htm');
  require('Includes/DpiMenu2.htm');
  require('Includes/DpiMenuAddAbout.htm');
?>

  <!-- Content of AboutDPI Page -->
<div align="left">

  <table border="0" cellpadding="1" cellspacing="0" id="table4" width="650">

  <tr>
    <td>

<table border="0" cellpadding="5" style="border-collapse: collapse"
width="100%" id="table6">
  <tr>
    <td width="45%" height="45" colspan="2">
      <br>
      <font face="Verdana" size="2"><b>Dutch Precision</b> is owned and operated
      by Barry Van Wechel, a Marine Corps veteran formerly from Northwest
      Iowa.</font><p style="text-align: left">
      <font face="Verdana" size="2">Barry started DPI in 1997, and bought
      Coastal Calipers
      Remanufacturing in 1994,
      expanding the business from Hudson, FL to it's current home in Port
      Richey.</font></p>
      <b><font face="Verdana" size="2">Directions to DPI</font></b><p
      style="text-align: left">
      <font face="Verdana" size="2">Port Richey is just East of highway 19,
      about
      28 miles north of Tampa.</font></p>
      <p style="text-align: left"><font face="Verdana" size="2">From Hwy 19,
      turn east on Richey Drive (524) and go one block to Leo Kidd.&nbsp;  
      Then turn south on Leo Kidd to Dasher Court - turn back west - DPI is on
      the south
      side of the road.</font></p> </td>
    <td width="52%" height="45" colspan="2">
      <p align="center" style="text-align: center">
      </td>
  </tr>

```



```

        background: white;
    }
    .footer {
        padding: 10px;
        margin: 0px;
        border-top: thin solid #000000;
        position: absolute;
        bottom: 0px;
        left: 0px;
    }

    p {color:black; font-size:12pt; text-align:justify;
        font-family:arial,sans-serif}
    p.foot {color:white; font-size:9pt; text-align:center;
        font-family:arial,sans-serif; font-weight:bold}
    a:link,a:visited,a:active {color:white}
    a:link,a:hover {color:white; text-decoration: none;}

    #onwhitelink {color:#006699; background: transparent;}
    #onwhitelink:visited {color: #006699;}
    #onwhitelink:hover {color: #333; text-decoration: none;}

    body {
        background: url("images/bk_RightSideDpi.gif");
        background-position: 680px 0px;
        background-repeat: repeat-y;
    }

    body.div {
        font face: Geremond;
        font color: #000080;
    }

</style>
</head>
<body>
<div align="left" border= "0" margin= "0 0 0 0">
<table width="650px" border= "0 0 0 0" margin= "0 0 0 0" cellpadding="0"
cellspacing="0" id="table2">
<tr>
<td>

        <p align="left">

            
        </td>
</tr>
</table>

</div>

<!-- Header -->
<html>
<head>

<!-- Image Rotation Script -->
    <SCRIPT LANGUAGE="JavaScript" TYPE="text/javascript">
<!--
// Courtesy of SimplytheBest.net - http://simplythebest.net/scripts/
gSlideshowInterval = 5;
gNumberOfImages = 4;

```

```

gImages = new Array(gNumberOfImages);
gImages[0] = "images/Caliper4ab.jpg";
gImages[1] = "images/Caliper4bb.jpg";
gImages[2] = "images/Caliper4cb.jpg";
gImages[3] = "images/Caliper4db.jpg";

function canManipulateImages() {
    if (document.images)
        return true;
    else
        return false;
}
function loadSlide(imageURL) {
    if (gImageCapableBrowser) {
        document.slide.src = imageURL;
        return false;
    }
    else {
        return true;
    }
}
function nextSlide() {
    gCurrentImage = (gCurrentImage + 1) % gNumberOfImages;
    loadSlide(gImages[gCurrentImage]);
}
gImageCapableBrowser = canManipulateImages();
gCurrentImage = 0;
setInterval("nextSlide()",gSlideshowInterval * 1000);
// -->
</SCRIPT>

<title>Dutch Precision Caliper Remanufacturing</title>
<style>
    h1 {color:white; font-size: 24pt; text-align:center;
        font-family:arial,sans-serif}
    .menu {color:white; font-size:12pt; text-align:center;
        font-family:arial,sans-serif; font-weight:bold}
    table {
        background: white;
    }
    .footer {
        padding: 10px;
        margin: 0px;
        border-top: thin solid #000000;
        position: absolute;
        bottom: 0px;
        left: 0px;
    }

    p {color:black; font-size:12pt; text-align:justify;
        font-family:arial,sans-serif}
    p.footer {color:white; font-size:9pt; text-align:center;
        font-family:arial,sans-serif; font-weight:bold}

    a:link,a:visited,a:active {color:white}
    a:link,a:hover {color:white; text-decoration: none;}

    #onwhitelink {color:#006699; background: transparent;}
    #onwhitelink:visited {color: #006699;}
    #onwhitelink:hover {color: #333; text-decoration: none;}

    body {
        background: url("images/bk_RightSideDpi.gif");

```



```

<tr >
  <td bgcolor="#5A3C03">
    <p align="left"><b><font face="Garamond" size="6"
color="#FFFFFF">&nbsp;Caliper
Remanufacturing</font></b></td>

    <td bgcolor="#5A3C03" valign="bottom">
      <p align="right" style="text-align: right">
        <font color="#FFFFFF" face="Garamond" size="2"><a
href="Products.php">Products</a> |
        <a href="PartsLookup.php">Parts Lookup</a> |
        <a href="AboutDPI.php">About DPI</a> |
        <a href="Index.php">Home</a></font></td>

</tr>
</table>

```

```

</div>

```

```

<!-- DpiMenuAddAbout -->

```

```

<div align="left">
  <table width="650" cellpadding="4" cellspacing="0" Top-border="1"
style="border-left-width: 2px;
border-right-width: 2px; " id="table3">
    <tr >
      <td bgcolor="#5A3C03">
        <b><font color="#FFFFFF" face="Verdana">&nbsp;- About DPI</font></b></td>

        <td bgcolor="#5A3C03" valign="bottom" width="265">
          <p align="right" style="text-align: right">&nbsp;</td>

    </tr>
  </table>
</div>

```

Appendix C: Database Coding

Included here are three representative database scripts to create and populate tables using a web browser interface.

1. CaliperTbl (DpCal_ID, Dpi_Num, Side, Cal_Descr, Casting_ID)

```
<?php
// Data functions for table CaliperTbl
// This script and data application were generated by AppGini 2.61

function insert()
{
    global $HTTP_SERVER_VARS, $HTTP_GET_VARS, $HTTP_POST_VARS,
    $HTTP_POST_FILES, $Translation;

    if(get_magic_quotes_gpc())
    {
        $Dpi_Num = $HTTP_POST_VARS["Dpi_Num"];
        $Side = $HTTP_POST_VARS["Side"];
        $Cal_Descr = $HTTP_POST_VARS["Cal_Descr"];
    }
    else
    {
        $Dpi_Num = addslashes($HTTP_POST_VARS["Dpi_Num"]);
        $Side = addslashes($HTTP_POST_VARS["Side"]);
        $Cal_Descr = addslashes($HTTP_POST_VARS["Cal_Descr"]);
    }
    if($Dpi_Num==""){
        echo StyleSheet() . "\n\n<div class=Error>" .
        $Translation["error:"] . " 'Dpi_Num': " . $Translation['field not null'] .
        "</div>";
        exit;
    }

    sql("insert into CaliperTbl (Dpi_Num, Side, Cal_Descr) values (" .
    (($Dpi_Num != "") ? "'$Dpi_Num'" : "NULL") . ", " . (($Side != "") ? "'$Side'"
    : "NULL") . ", " . (($Cal_Descr != "") ? "'$Cal_Descr'" : "NULL") . ")");
    return mysql_insert_id();
}

function delete($selected_id)
{
    // insure referential integrity ...
    global $Translation;

    // child table: CalipModelJct
    $res = sql("select DpCal_ID from CaliperTbl where
    DpCal_ID='$selected_id'");
    $DpCal_ID = mysql_fetch_row($res);
```

```

    $rires = sql("select count(1) from CalipModelJct where
DpCal_ID='$DpCal_ID[0]');
    $rirow = mysql_fetch_row($rires);
    if($rirow[0])
    {
        $RetMsg = $Translation["couldn't delete"];
        $RetMsg = str_replace("<RelatedRecords>", $rirow[0], $RetMsg);
        $RetMsg = str_replace("<TableName>", "CalipModelJct", $RetMsg);
        return $RetMsg;
    }

    sql("delete from CaliperTbl where DpCal_ID='$selected_id");
}

function update($selected_id)
{
    global $HTTP_SERVER_VARS, $HTTP_GET_VARS, $HTTP_POST_VARS, $Translation;

    if(get_magic_quotes_gpc())
    {
        $Dpi_Num = $HTTP_POST_VARS["Dpi_Num"];
        $Side = $HTTP_POST_VARS["Side"];
        $Cal_Descr = $HTTP_POST_VARS["Cal_Descr"];
    }
    else
    {
        $Dpi_Num = addslashes($HTTP_POST_VARS["Dpi_Num"]);
        $Side = addslashes($HTTP_POST_VARS["Side"]);
        $Cal_Descr = addslashes($HTTP_POST_VARS["Cal_Descr"]);
    }

    sql("update CaliperTbl set " . "Dpi_Num=" . (($Dpi_Num != "") ?
"'$Dpi_Num'" : "NULL") . ", " . "Side=" . (($Side != "") ? "'$Side'" : "NULL")
. ", " . "Cal_Descr=" . (($Cal_Descr != "") ? "'$Cal_Descr'" : "NULL") . "
where DpCal_ID='$selected_id");
}

function form($selected_id = "", $AllowUpdate = 1, $AllowInsert = 1,
$AllowDelete = 1)
{
    // function to return an editable form for a table records
    // and fill it with data of record whose ID is $selected_id. If
    $selected_id
    // is empty, an empty form is shown, with only an 'Add New'
    // button displayed.

    global $Translation;

    if(!$AllowInsert && $selected_id==""){ return ""; }

    $code = "<br><table border=1 bordercolor=navy cellpadding=0
cellspacing=0><tr><td><div class=TableTitle>Detail
View</div></td></tr><tr><td><table>";
    $code .= "\n<tr><td colspan=2></td><td rowspan=7 valign=top>";
    if($AllowInsert)
        $code .= "<div><input type=image src=insert.gif name=insert alt='"
. $Translation["add new record"] . "'></div>";

    if($selected_id)
    {
        $res = sql("select * from CaliperTbl where
DpCal_ID='$selected_id");
    }

```



```

        $Make_Abbv = addslashes($HTTP_POST_VARS["Make_Abbv"]);
    }
    if($Make_Name==""){
        echo StyleSheet() . "\n\n<div class=Error>" .
$Translation["error:"] . " 'Make_Name': " . $Translation['field not null'] .
"</div>";
        exit;
    }

    sql("insert into MakeTbl (Make_Name, Make_Abbv) values (" . (($Make_Name
!= "") ? "'$Make_Name'" : "NULL") . ", " . (($Make_Abbv != "") ? "'$Make_Abbv'"
: "NULL") . ")");
    return mysql_insert_id();
}

function delete($selected_id)
{
    // insure referential integrity ...
    global $Translation;

    // child table: ModelTbl
    $res = sql("select Make_ID from MakeTbl where Make_ID='$selected_id'");
    $Make_ID = mysql_fetch_row($res);
    $rires = sql("select count(1) from ModelTbl where
Make_ID='$Make_ID[0]'");
    $rirow = mysql_fetch_row($rires);
    if($rirow[0])
    {
        $RetMsg = $Translation["couldn't delete"];
        $RetMsg = str_replace("<RelatedRecords>", $rirow[0], $RetMsg);
        $RetMsg = str_replace("<TableName>", "ModelTbl", $RetMsg);
        return $RetMsg;
    }

    sql("delete from MakeTbl where Make_ID='$selected_id'");
}

function update($selected_id)
{
    global $HTTP_SERVER_VARS, $HTTP_GET_VARS, $HTTP_POST_VARS, $Translation;

    if(get_magic_quotes_gpc())
    {
        $Make_Name = $HTTP_POST_VARS["Make_Name"];
        $Make_Abbv = $HTTP_POST_VARS["Make_Abbv"];
    }
    else
    {
        $Make_Name = addslashes($HTTP_POST_VARS["Make_Name"]);
        $Make_Abbv = addslashes($HTTP_POST_VARS["Make_Abbv"]);
    }

    sql("update MakeTbl set " . "Make_Name=" . (($Make_Name != "") ?
"'$Make_Name'" : "NULL") . ", " . "Make_Abbv=" . (($Make_Abbv != "") ?
"'$Make_Abbv'" : "NULL") . " where Make_ID='$selected_id'");
}

function form($selected_id = "", $AllowUpdate = 1, $AllowInsert = 1,
$AllowDelete = 1)
{
    // function to return an editable form for a table records
    // and fill it with data of record whose ID is $selected_id. If
    $selected_id

```

```

// is empty, an empty form is shown, with only an 'Add New'
// button displayed.

global $Translation;

if(!$AllowInsert && $selected_id==""){ return ""; }

$code = "<br><table border=1 bordercolor=navy cellpadding=0
cellspacing=0><tr><td><div class=TableTitle>Detail
View</div></td></tr><tr><td><table>";
$code .= "\n\t<tr><td colspan=2></td><td rowspan=7 valign=top>";
if($AllowInsert)
    $code .= "<div><input type=image src=insert.gif name=insert alt='"
. $Translation["add new record"] . "'></div>";

if($selected_id)
{
    $res = sql("select * from MakeTbl where Make_ID='$selected_id'");
    $row = mysql_fetch_array($res);

    $code .= "<br>";
    if($AllowUpdate)
        $code .= "<div><input type=image src=update.gif vspace=1
name=update alt='" . $Translation["update record"] . "'></div>";
    if($AllowDelete)
        $code .= "<div><input type=image src=delete.gif vspace=1
name=delete alt='" . $Translation["delete record"] . "' onclick='return
confirm(\"" . $Translation["are you sure?"] . "\");'></div>";
        $code .= "<div><input type=image src=deselect.gif vspace=1
name=deselect alt='" . $Translation["deselect record"] . "'></div>";
    }

$code .= "</td></tr>";

// Detail view form fields
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Make_ID</div></td><td class=TableBody width=300>" .
$row['Make_ID'] . "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'><b class='Error'>" . $Translation['*'] . "</b>
Make_Name</div></td><td class=TableBody width=300><input size= 40 type=text
class=TextBox name=Make_Name value='" . htmlspecialchars($row['Make_Name'],
ENT_QUOTES) . "'>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Abbv</div></td><td class=TableBody width=300><input
size=10 type=text class=TextBox name=Make_Abbv value='" .
htmlspecialchars($row['Make_Abbv'], ENT_QUOTES) . "'>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td></tr>";

$code .= "</table></td></tr></table>";

return $code;
}
?>

```

3. ModelTbl (Model_ID, Model_Name, Make_ID, YrRange_ID, Sub_ID)

```

<?php
// Data functions for table ModelTbl

```

```
// This script and data application were generated by AppGini 2.61
```

```
function insert()
{
    global $HTTP_SERVER_VARS, $HTTP_GET_VARS, $HTTP_POST_VARS,
    $HTTP_POST_FILES, $Translation;

    if(get_magic_quotes_gpc())
    {
        $Model_Name = $HTTP_POST_VARS["Model_Name"];
        $Make_ID = $HTTP_POST_VARS["Make_ID"];
        $Year_ID = $HTTP_POST_VARS["Year_ID"];
        $Sub_ID = $HTTP_POST_VARS["Sub_ID"];
    }
    else
    {
        $Model_Name = addslashes($HTTP_POST_VARS["Model_Name"]);
        $Make_ID = addslashes($HTTP_POST_VARS["Make_ID"]);
        $Year_ID = addslashes($HTTP_POST_VARS["Year_ID"]);
        $Sub_ID = addslashes($HTTP_POST_VARS["Sub_ID"]);
    }
    if($Model_Name== ""){
        echo StyleSheet() . "\n\n<div class=Error>" .
        $Translation["error:"] . " 'Make_Name': " . $Translation['field not null'] .
        "</div>";
        exit;
    }

    sql("insert into ModelTbl (Model_Name, Make_ID, Year_ID, Sub_ID) values
    (" . (($Model_Name != "") ? "'$Model_Name'" : "NULL") . ", " . (($Make_ID !=
    "") ? "'$Make_ID'" : "NULL") . ", " . (($Year_ID != "") ? "'$Year_ID'" :
    "NULL") . ", " . (($Sub_ID != "") ? "'$Sub_ID'" : "NULL") . ")");
    return mysql_insert_id();
}

function delete($selected_id)
{
    // insure referential integrity ...
    global $Translation;

    // child table: CalipModelJct
    $res = sql("select Model_ID from ModelTbl where
    Model_ID='$selected_id'");
    $Model_ID = mysql_fetch_row($res);
    $rires = sql("select count(1) from CalipModelJct where
    Model_ID='$Model_ID[0]'");
    $rirow = mysql_fetch_row($rires);
    if($rirow[0])
    {
        $RetMsg = $Translation["couldn't delete"];
        $RetMsg = str_replace("<RelatedRecords>", $rirow[0], $RetMsg);
        $RetMsg = str_replace("<TableName>", "CalipModelJct", $RetMsg);
        return $RetMsg;
    }

    sql("delete from ModelTbl where Model_ID='$selected_id'");
}

function update($selected_id)
{
    global $HTTP_SERVER_VARS, $HTTP_GET_VARS, $HTTP_POST_VARS, $Translation;

    if(get_magic_quotes_gpc())
```



```

{
    $Model_Name = $HTTP_POST_VARS["Model_Name"];
    $Make_ID = $HTTP_POST_VARS["Make_ID"];
    $Year_ID = $HTTP_POST_VARS["Year_ID"];
    $Sub_ID = $HTTP_POST_VARS["Sub_ID"];
}
else
{
    $Model_Name = addslashes($HTTP_POST_VARS["Model_Name"]);
    $Make_ID = addslashes($HTTP_POST_VARS["Make_ID"]);
    $Year_ID = addslashes($HTTP_POST_VARS["Year_ID"]);
    $Sub_ID = addslashes($HTTP_POST_VARS["Sub_ID"]);
}

    sql("update ModelTbl set " . "Model_Name=" . (($Model_Name != "") ?
    "'$Model_Name' : 'NULL') . ", " . "Make_ID=" . (($Make_ID != "") ?
    "'$Make_ID' : 'NULL') . ", " . "Year_ID=" . (($Year_ID != "") ? "'$Year_ID' :
    'NULL') . ", " . "Sub_ID=" . (($Sub_ID != "") ? "'$Sub_ID' : 'NULL') . " where
    Model_ID='$selected_id'");
}

function form($selected_id = "", $AllowUpdate = 1, $AllowInsert = 1,
$AllowDelete = 1)
{
    // function to return an editable form for a table records
    // and fill it with data of record whose ID is $selected_id. If
    $selected_id
    // is empty, an empty form is shown, with only an 'Add New'
    // button displayed.

    global $Translation;

    if(!$AllowInsert && $selected_id==""){ return ""; }

    $code = "<br><table border=1 bordercolor=navy cellpadding=0
    cellspacing=0><tr><td><div class=TableTitle>Detail
    View</div></td></tr><tr><td><table>";
    $code .= "\n\t<tr><td colspan=2></td><td rowspan=7 valign=top>";
    if($AllowInsert)
        $code .= "<div><input type=image src=insert.gif name=insert alt='"
        . $Translation["add new record"] . "'></div>";

    // combobox: Make_ID
    $combo_Make_ID = new DataCombo;
    $combo_Make_ID->Query = "select Make_ID, Make_Name from MakeTbl order by
    Make_Name";
    $combo_Make_ID->SelectName = "Make_ID";
    // combobox: Year_ID
    $combo_Year_ID = new DataCombo;
    $combo_Year_ID->Query = "select Year_ID, YrRange from YearTbl order by
    YrRange";
    $combo_Year_ID->SelectName = "Year_ID";
    // combobox: Sub_ID
    $combo_Sub_ID = new DataCombo;
    $combo_Sub_ID->Query = "select Sub_ID, SubValue from SubMoTbl order by
    SubValue";
    $combo_Sub_ID->SelectName = "Sub_ID";

    if($selected_id)
    {
        $res = sql("select * from ModelTbl where
    Model_ID='$selected_id'");
        $row = mysql_fetch_array($res);
    }
}

```

```

$combo_Make_ID->SelectedData = $row["Make_ID"];
$combo_Year_ID->SelectedData = $row["Year_ID"];
$combo_Sub_ID->SelectedData = $row["Sub_ID"];

$code .= "<br>";
if($AllowUpdate)
    $code .= "<div><input type=image src=update.gif vspace=1
name=update alt='\" . $Translation["update record"] . \"'></div>";
    if($AllowDelete)
        $code .= "<div><input type=image src=delete.gif vspace=1
name=delete alt='\" . $Translation["delete record"] . \"' onclick='return
confirm(\"\" . $Translation["are you sure?"] . \"\");'></div>";
        $code .= "<div><input type=image src=deselect.gif vspace=1
name=deselect alt='\" . $Translation["deselect record"] . \"'></div>";
    }
    $combo_Make_ID->Render();
    $combo_Year_ID->Render();
    $combo_Sub_ID->Render();

$code .= "</td></tr>";

// Detail view form fields
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Model_ID</div></td><td class=TableBody width=300>" .
$row['Model_ID'] . "&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'><b class='Error'>" . $Translation['*'] . "</b>
Make_Name</div></td><td class=TableBody width=300><textarea class=TextBox
name=Model_Name cols=50 rows=3>" . htmlspecialchars($row["Model_Name"],
ENT_QUOTES) . "</textarea>&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Make_ID</div></td><td class=TableBody
width=300>$combo_Make_ID->HTML&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Year_ID</div></td><td class=TableBody
width=300>$combo_Year_ID->HTML&nbsp;&nbsp;&nbsp;</td></tr>";
$code .= "\n\t<tr><td class=TableHeader valign=top><div class=TableHeader
style='text-align:right;'>Sub_ID</div></td><td class=TableBody
width=300>$combo_Sub_ID->HTML&nbsp;&nbsp;&nbsp;</td></tr>";

$code .= "</table></td></tr></table>";

return $code;
}
?>

```

Appendix D: Website Image Design Examples

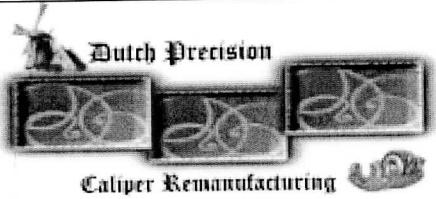
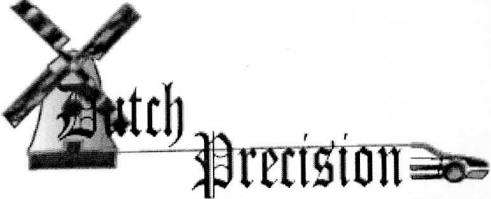
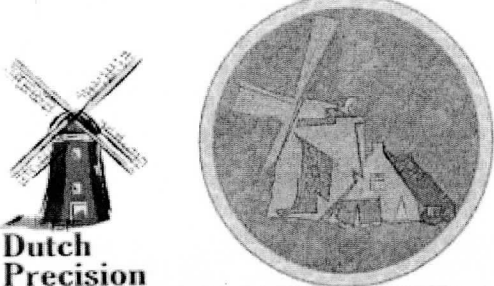
	
	
	
	
	
	
	

Table D-1: Website Image Designs

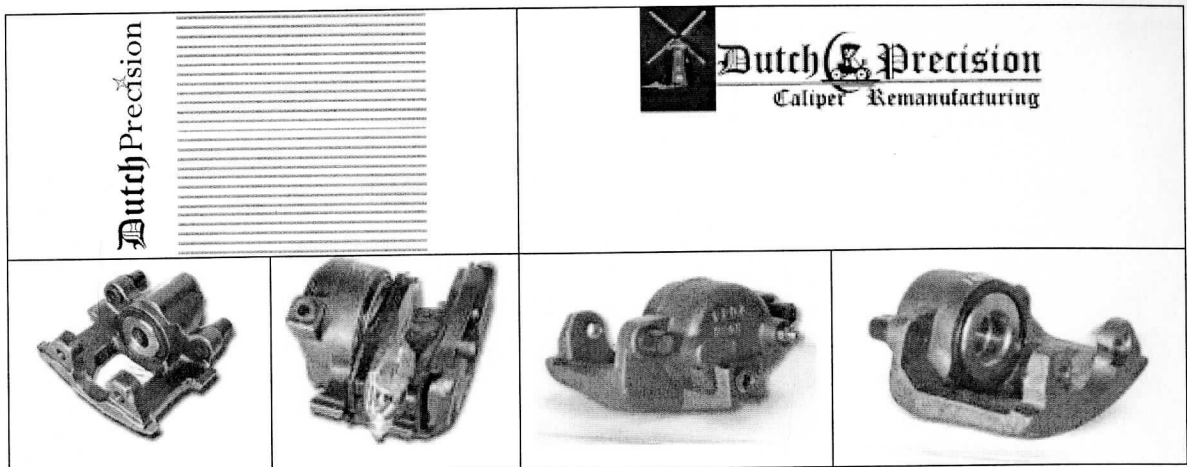


Table D-2: Website Image Designs Cont.

The images above are examples of the range of imagery ideas for logos, web page frames and promotional photography.

Appendix E – Pictures of Dutch Precision, Inc.



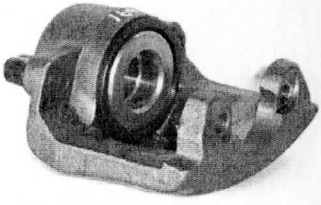

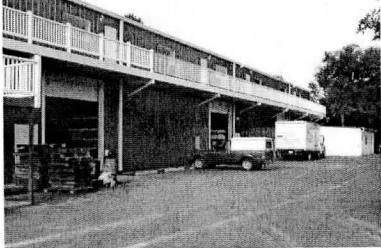
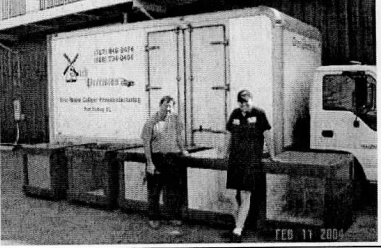

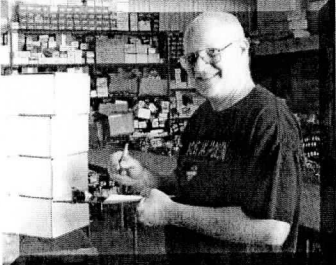

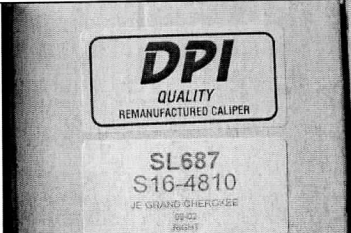
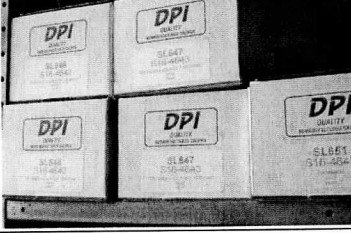
		
<p style="text-align: center;">Owner: Barry Van Wechel</p>	<p style="text-align: center;">DPI Building Sign (DPI no longer rebuilds axles)</p>	<p style="text-align: center;">Typical Brake Caliper</p>
		
<p style="text-align: center;">Core Return Entrance</p>	<p style="text-align: center;">DPI Facility</p>	<p style="text-align: center;">Deliver Route Day</p>
		
<p style="text-align: center;">Alvin</p>	<p style="text-align: center;">Steve</p>	<p style="text-align: center;">Scott</p>
		
<p style="text-align: center;">Caliper Label</p>		<p style="text-align: center;">Caliper Boxes</p>

Table E-1: Pictures Dutch Precision, Inc.