

**2009**

**University of North Carolina Wilmington**  
**Master of Science in**  
**Computer Science and Information Systems**  
**Proceedings**

**<https://csbapp.uncw.edu/mscsis>**

A LOW-COST TELE-OPERATED ROBOTIC PLATFORM

Robert M. Harrison, Jr.

A Capstone Project Submitted to the  
University North Carolina Wilmington in Partial Fulfillment  
of the Requirements for the Degree of  
Master of Science

Department of Computer Science  
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2009

Approved by

Advisory Committee

---

Dr. Bryan Reinicke

---

Dr. Tom Janicki

---

Dr. Curry Guinn, Chair

# Table of Contents

1. Introduction
2. Review and Analysis
  - 2.1 Motivations
  - 2.2 Supporting Academic Research
  - 2.3 Early Prototypes
    - 2.3.1. RC Prototype
    - 2.3.2. Turret
    - 2.3.3. Untethered Prototype
    - 2.3.4. Software Architecture
  - 2.4. Barriers and Breakthroughs
  - 2.5 Market Opportunity
    - 2.5.1. Market Analysis
    - 2.5.2. Available Products
    - 2.5.3. Available Resources
3. Competition
  - 3.1. Competitor Analysis
    - 3.1.1. Company Comparisons
    - 3.1.2. Product Comparisons
  - 3.2. Opportunity Analysis
4. Methodology and Plan
  - 4.1. Product Description

4.1.1. Technology

4.1.2. Business

4.1.3. Product Uniqueness

4.2. Strategy and Operations

4.3. Management

4.4. Financials

5. Conclusion

6. Future Work

7. References

## 1. Introduction

In 2007, Bill Gates wrote an article for *Scientific American* where he expressed his excitement for an emerging robotics industry. He paralleled the infancy years of the computer industry of the 1970's to the present-day robotics community. Back then, machines were large, were far from elegant, and were only used for specialized roles. Once a core group of individuals came together the movement gained surprising momentum. Mr. Gates went on to say that the lack of standard platforms to work from has impeded the growth of the robotics industry and that, “[w]henever somebody wants to build a new robot, they usually have to start from square one”. In the past, the most expensive aspect of robot development was the cost of hardware, but thanks to innovations in technology, we are able to acquire hardware at a fraction of the cost in comparison to years past. Red Whitaker, the leader of Carnegie Mellon University’s autonomous robot development team, said that, “the hardware capability is mostly there; now the issue is getting the software right”. [21]

Robotics is a growing area of research that has great potential, but unfortunately, the time and resources necessary to build a working robotic platform often eclipses the research for the intended platform. It is not uncommon for a robotics project to take years as well as many thousands of dollars. This is not an ideal scenario, but unfortunately options can be sparse. Developers who

undertake the task of developing their own platform can become mired in the interdisciplinary knowledge necessary for building a robot including software engineering, computer hardware, mechanical engineering, and electrical engineering. Having all the requisite skills necessary to develop a robot is a significant barrier for a beginner, and any interested party must make a decision on whether to build or buy a robot. There are two significant disadvantages to using currently available robotic development packages as starting points: 1). These packages often use proprietary hardware and software, which limits configurability of the robot and also limits the resources available for further development. 2). The off-the-shelf “starter kits” are prohibitively expensive. Because of the few number of vendors of such kits, there are few options available for the robot enthusiast. Robots that operate on the ground are nothing new, and have been shown to be effective by NASA and military developers alike. Both of these groups, of course, have vast and well-funded research and development departments that can make this a reality. This is not the case for academic researchers, hobbyists, and developers from small companies. I propose a platform that not only is cost effective but also is easy to develop and deploy using web technologies. [21]

The project will focus on two areas: prototype development and business analysis. First, the project proposal will demonstrate a prototype of the robot in conjunction with an implementation of a database-driven concurrency architecture. This concurrency architecture stores all pertinent control signals for the robot in a central data warehouse so that multiple applications can have

accurate, up-to-date data corresponding to the robot's hardware. Second, the project proposal will include an analysis of the current market, as well as delineate the robot's features and inner workings. This analysis will show deficiencies in the current marketplace and will provide points of comparison between available platforms and my current prototype. The finished capstone project will culminate with the design and development of a robot that is capable of tele-operated and autonomous control. The project goal is to develop and demonstrate a cost effective alternative to currently available robotics development platforms. To accomplish this, the robot will be developed from readily available parts intertwined with custom mechanical hardware all being controlled by a unique web architecture. The majority of the robot will consist of COTS (commercial-off-the-shelf) parts, and for the parts that aren't readily available, some custom parts will need to be made.

The finished capstone project will also include a business plan for a robotics development company named Axiom Robotics. This name is fitting since an axiom is a rational starting point from whence a logical progression of principles is applied. With this system, a developer will be able to start with a strong robotic development platform and derive greater and more complex robots from it.

## 2. Review and Analysis

### 2.1. Motivations

As mentioned previously, the robotics community is filled with purpose-built robots that are designed for specific tasks. While there are a myriad of applications for robots as well as a diverse number of robots designed to address those areas, it is challenging to find a robot that is versatile. Robots are used in the home, in industrial complexes, by the military, by law enforcement, by researchers, and by hobbyists. The robotics industry is obviously a broad clustering of robots developed around problem domains. To narrow the scope of this endeavor, my interests in robotics are relegated to Unmanned Ground Vehicles and the interaction of the robot with human and AI guidance. [20][22]

Robotic vehicles tend to be controlled in two basic ways: either a user decides each and every movement that the robot makes, or a computer system built around artificial intelligence decides the robot's movements. There is however a third method of control, and that is *semi-autonomous*, where a user makes higher-level decisions while the robot's AI controls the more mundane tasks. My motivation for developing a robotic development platform comes from a lack of available ground vehicles that work out of the box with little to no configuration and have tele-operation built-in. A robotic platform is a robotic vehicle that has user extensible features such that the vehicle is modifiable by using built-in hardware and software functions and behaviors. In contrast, a robotic vehicle may not have the necessary programmatic or hardware expandability to support the desired functionality of the user. The notion of a

“platform” or a base to develop from is crucial in the expansion and development of new robotic functionality and is crucial for putting tools in a user’s hands.

[1][2][3]

Few fully functioning robotic platforms are available for purchase, especially ones that are less than \$2,500. Most require extra components and know-how to make them work, and many times the platforms are small and complicated to integrate. Axiom Robotics' system would reduce the development time that would be needed to have a fully operational platform to accomplish the desired functionality of the customer. This goal can be realized by the development of a relatively low-cost yet robust robotic system that can be used or further developed depending on the functionality desired. [28]

Even if a developer found an inexpensive platform, there are a myriad of interdisciplinary facets that one would need to learn to be proficient. Many times developers must create their own platform in order to develop their original idea. As an additional complication many platforms are designed to be used in a closed environment where hardware and software changes are limited which prevents the developer from expanding the platform to meet their needs. There is a growing need for robots that are autonomous but also have an implementation that allows control from a distance. This control of a robot where the robot and operator are in different locations is referred to as tele-operated robotics, and the fusion of autonomy and tele-operation is referred to as semi-autonomous robotics.

The majority of robotic platforms focus on the procedural running of software, which greatly reduces the interactivity of the robot with its user. In some domains in the field of robotics there is a large focus on the AI of robotic systems, which has many benefits and ultimately is the goal of many robotic projects. However the available technology is not yet ready for the majority of robots to be fully autonomous, so many need human interactions. Axiom Robotics' goal is to bridge this gap by creating a platform that is tele-operated and has autonomous elements incorporated. This will allow the robot to move closer and closer to the holy grail of a fully autonomous system in a gradual and controlled manner. This introduces a whole new set of problems for providing a strong human-robot interface to facilitate this control that is very dependent on the concurrency of pertinent control signals. This platform will be one of the few out-of-box robots that can be controlled via tele-operation yet still have the underpinnings for a developer to add AI to it. Axiom Robotics will incorporate a web application into the robot to facilitate tele-operation control.

The complexity of creating a dynamic web-enabled human-computer interface to guide a robot that is in a different physical location than the operator is definitely a complex task but quite doable when web technologies are incorporated. Such technologies as high-speed databases, in-browser Java applets, and hypertext pre-processing languages have evolved to the point that live applications and objects can be controlled through a web browser. Robots are one such technology that can benefit from this interaction, since the aforementioned technologies are relatively cheap and prototypes can be rapidly

built. By utilizing web technologies, the developer gains from the web's robust architecture, cross-platform compatibility, and cost effectiveness, which ultimately translates into a fertile development environment.

To illustrate how web technologies handle many of the complex tasks of robot development, here is a simple example. Creating an interface for a robot is no small task since one would need to develop a way to acquire data and video, render that data and video to the screen, and provide a method of control that has some feedback. To accomplish this one would need to develop a networking architecture, a data concurrency hierarchy, a user interface that renders all of the on-screen elements, and an input device such as a joystick. This interface also needs to be platform independent to reduce the complications of a closed system approach. This task now sounds very complex until web technologies are used. Since the web is built upon established networking protocols such as the IP and HTTP standards, much of the complications subside. By utilizing a database, multiple users can interact with the data within their browsers. Using an HTML layout for an interface solves the problem of not only cross-platform compatibility but also the development of interfaces since the rendering of on-screen graphics is handled by the browser. The web browser even handles the security settings of interacting with these elements. This simple example shows how many development hurdles are easily solved by using the web.

## **2.2. Supporting Academic Research**

There are many useful applications of tele-operated robotics. Ambrose and Spain (2004) discussed how “[t]asks that currently require people to work in

dangerous arenas are perfect candidates for humanoid robots.” Now while the unmanned ground vehicle used in this research is not humanoid, it still will be able to perform tasks that its operator may not wish to be present for. In a way, a tele-operated robot is an abstraction of control from the operator to the task at hand, and this is why NASA and DARPA have both utilized tele-operated robotics. Outer space and the battlefield are both places that could use tele-operated robots to ensure the safety of their controller. Another interesting point that Ambrose and Spain bring up is the time lag inherent in tele-operation. Since the operator and robot are in different locations, there is a transmittal time difference from when a command is sent from the operator to the robot and back again. They conclude, however, that this time lag is not really a problem, since the operator can slow his/her movements. [12] Sometimes slowing ones movement is not acceptable, and the robot must begin to operate itself in the presence of sparse control inputs. This notion gives rise to the concept of an Agent, a computer “driver” that actually controls the movements of the robot, and merely accepts the operator’s input as a guide, rather than an absolute control signal. [14] This delay issue is also referred to as control feedback latency. Since the Agent in the presence of infrequent control updates must drive itself, the fusing of other inputs and controls is essential in determining the next move for the vehicle. For instance, a distance sensor can be used to detect obstacles that may hinder the robot. The Agent would recognize the obstacle so that it could either drive around it using some predefined logic, or it could stop and wait for input from the operator. With further refinement, the robot could actually adapt to

such a control bare situation rather than just have preprogrammed tactics. What the robot must consider is position, orientation and speed before a choice can be made, which is also referred to as the anticipated intrinsic state of the vehicle, rather than the currently available and outdated position, orientation and speed. This choice must coincide with the wishes of the operator, and this interaction must be defined in such a way that the operator and Agent know which inputs have priority. [14]

Since an Agent is designed to make decisions in the absence of direct control, it stands to reason that more and more functions of the robot could be turned over to the Agent instead of having to wait on the operator to make control decisions. This gradual increase in autonomy leads to the ultimate goal of this research, to create a semi-autonomous UGV that can navigate itself through certain situations, but may need input for more complex situations. This concept is commonly referred to as human-robot interaction (HRI).

Shiomi et al (2008) describes differing layers in a semi-autonomous robot such as the behavioral layer and the reflective layer. The behavioral layer is likened to the subconscious movements of mammals, where the action is completed without a decision needing to be made. This behavior is acquired in mammals through repetition, which must be incorporated into the robot to accomplish a similar effect. The reflective layer on the other hand is the layer that the operators interact with, and this interaction is carried down to the behavioral layer. By gradually automating the behavioral layer, fewer actions are necessary

at the reflective layer. Some of the initial problem domains for partial autonomy come in the form of object avoidance and goal-directed locomotion. [13]

Parasuraman and Sheridan (2000) delve further into levels of automation in their seminal paper “A Model for Types and Levels of Human Interaction with Automation.” The authors delineate four broad classes that are subject to automation: 1) information acquisition; 2) information analysis; 3) decision and action selection; 4) action implementation. Within these levels there can be a continuum of automation from low to high. Referring back to the concept of the Agent, it depends on whether the system is an open loop or closed loop control system. The open loop would allow a user to make decisions before the Agent’s decision is executed while the closed loop system would execute the autonomous action without waiting on the user input. [3]

Chen and Barnes (2008) expand the semi-autonomous notion for military multi-tasking robotic environments, where automatic target acquisition can greatly reduce the amount of time that an operator would need to spend on target acquisition. The authors go on to explain how a divided attention paradigm necessitates the robot being capable of acquiring targets without user input when a single user is commanding multiple robots. It was also pointed out that a combination of semi-autonomous and tele-operated control would be necessary to garner the most effective target acquisition quotient. In essence, both the robot and operator would find different targets, and the fusion of both yielded the most targets acquired. [1]

Finally, as a corollary to Chen and Barnes, Olsen and Wood (2004) state that the “goal of human-robot interaction is to allow one user to operate multiple robots simultaneously. In such a scenario the robots provide leverage to the user’s attention.” The authors go on to say that fully autonomous robots will not be a reality in the foreseeable future, and that human-directed robots are the key to reaching full autonomy. This will be useful for robots of all different intelligence levels where a single user interface can be used to instruct them. It is evident that a one-to-one ratio of robot-to-operator would be inefficient since the operator could under normal circumstances perform the same task. Conversely, if a single operator could control multiple robots, then more work is being accomplished over the work that could be accomplished by the operator alone. [2]

## **2.3. Early Prototypes**

### **2.3.1. RC Prototype**

This project began as a chassis kit that consisted of two plastic decks, and two motor-driven sets of tracks. After searching among various manufacturers for the appropriate electronics, a radio controlled tank-like treaded platform emerged. The electronics for this machine were very minimal, consisting of a battery, control receiver, and a motor controller all being controlled using an R/C airplane transmitter. Later on I decided to build a larger and stronger set of aluminum chassis plates to allow for expansion. By doing so, the robot was able to handle heavier loads, and had more space to mount electronics. At that point,

the development of the robot was split into two distinct sections: a turret and a base.

### **2.3.2. Turret**

The upper half consisted of a two-axis camera mount that allowed panning and tilting of a camera. A wireless IP camera was mounted in the pan/tilt unit, and could display full motion video over the network, which was displayed in a web browser. To finish the upper half implementation, a servo motor controller was included. The purpose of the servo motor controller was to provide precise movement to the pan/tilt unit by interfacing a USB to Pulse Width Modulation conversion facilitated by the microcontroller. By specifying a coordinate pair in the software, the servos could rotate the camera into the position that was specified. This servo microcontroller was one of the most instrumental devices on the robot, for without it, it would be very difficult to move the camera into position.

**Appendix A: Figure 1 shows the Java Agent Diagram and how the microcontrollers interact with software and Figure 6 shows the actual microcontrollers being used.**

Included in the upper half of the robot was also a power source. This was accomplished by a 9.6-volt nickel metal hydride rechargeable battery pack that was directly connected to the servo microcontroller to power the servomotors. There was also an indirect power source of 5 volts, which came from a power converter that takes the 9.6 volts from the battery pack, and converts it to a stable 5 volts of power. This stable 5 volts was necessary for powering the IP camera.

I considered this a milestone in the development process, since I had the necessary components to rotate a camera and see its live output. This allowed me to focus on the lower half of the robot that provided the locomotion necessary to move the camera from location to location.

### **2.3.3. Untethered prototype**

At this point, I had a working tele-operated turret, and I needed not only a method of locomotion but also a control structure to command the upper and lower halves at a distance. I used the R/C base unit and modified it with a motor microcontroller that would give me a software-driven command structure similar to the servo microcontroller command structure. This controller on the tracked base controlled each motor for each track. The motor controller converted the inputted signals from software into left and right electrical power outputs that drove the motors that drove the treads.

#### **Appendix A: Figure 5 shows a class diagram of the Java Agent.**

The robot also needed more than just vision for sensing its environment, so I added a sensor microcontroller to poll a set of infrared distance sensors. This gave the robot spatial data to extrapolate where objects might be which would allow it to avoid them. Finally, the last piece of hardware needed for this project was a small form factor laptop. It is evident that the power, size, and versatility of today's netbook PCs are sufficient to control each microcontroller yet still be small enough to fit on the robot itself. Since all of the microcontrollers are USB, the netbook could reside on top of the robot chassis and utilize the microcontrollers to navigate the robot.

Since the robot, with the hardware listed above, had locomotion, vision, sensing capabilities, and wireless connectivity, it could be controlled without being tethered and without direct supervision. In other words the robot could be completely tele-operated from a distance. The operator or client could be operated in one physical location while the robot or server could be operated in a separate physical location.

The untethered prototype was tested during the summer of 2009 and I was able to drive the robot around the inside of a building without direct supervision. While this was an exciting development, it also brought to light all of the aspects of the robot that needed to be improved. These improvements are addressed in the Barriers and Breakthroughs section.

#### **2.3.4. Software Architecture**

Since the goal of this project is to tele-operate a vehicle utilizing web technologies, another technology layer is needed. First, there needs to be a central repository of data that defines each aspect that is to be controlled or monitored. For instance, the position of the pan/tilt unit could be represented as a coordinate pair in a database. The human operator could update this coordinate pair in the database, and then the server could read this coordinate pair from the database and update the pan/tilt unit's coordinates accordingly.

The software developed to accomplish this framework is divided into two parts: the client and the server. The client is a web application that has two distinct features. The first being a window that views what the camera sees and displays this video in real time. The other part of the web application is the

control unit. This takes user input via directional arrows and updates the database accordingly.

**Appendix A: Figure 3 shows the client side of the web interaction and Figure 4 shows the server side of the web interaction.**

The server side consists of a MySQL database that stores all of the coordinates and other pertinent information regarding the robot. The reason for implementing a database is the speed and atomicity of transactions. The database can query for data faster rate than the Agent can request it. Also, by having each read and update to the database being done in an atomic way, there is no need to worry about deadlock. An Apache web server was added to handle all of the user interaction of the client-side web application and melds the client's application to the database. There is also a Java "Driver" that polls the database at regular intervals to request the latest coordinates for the robot. Once the Driver has the new coordinates, it interfaces with each microcontroller to update each one with the current values.

**Appendix B shows the structure for the database.**

Finally, the last piece of the server is the network camera itself. Since the IP camera is a small web server itself, it was unnecessary for the robot server to handle the camera's server operations. The client's viewer directly connects with the camera's server output via a Java applet that is transferred from the camera server to the client. A separate router was needed to isolate the network traffic and allow for mobility of the network. During experimentation it was noticed that

the throughput of the router needed to be high to accommodate the bandwidth needs of the IP camera.

## **2.4. Barriers and Breakthroughs**

The project had its successes and blunders. The camera was originally used on the UNCW campus using the university's network for viewing the web cam. At first, the Computer Science's server "Torvalds" was used for hosting the database as well as website. The server was later pushed down to the robot itself, since this would allow the robot to make local calls to the database, which are much faster than relying on the Torvalds database. By hosting the web server on the robot, it helped get around networking issues that were preventing the robot from connecting to outside data sources from within the UNCW network.

As mentioned before, due to degraded campus network speeds, it was necessary to acquire a standalone router that would allow for a closed and uncluttered network that could manage. This new network allowed for speed, flexibility, and manageability that couldn't be attained using the campus network.

I was disappointed with two aspects of the camera, both of which were hardware related. The original pan unit was providing inaccurate rotations of the camera, and I suspect that the variable rate potentiometer, which determines exact servo rotation, was not reading properly. This unit has subsequently been replaced with a simpler direct drive unit that has been working with much greater accuracy and a lower cost than the previous unit. I have also noticed an increase in speed since switching to the direct drive pan unit. It also offers more hardware

flexibility by allowing different servos to be installed in the pan unit to meet speed and torque demands.

The original servo microcontroller card that I was using was limited in the sense that it could only control four servos, and each servo had to be moved individually. I originally thought that simultaneous movements via the servo controller were not possible with the current hardware. I eventually opted for a more advanced servo controller unit that gave me more features and provided much smoother movements.

## **2.5. Market Opportunity**

It is evident that the robotics industry is growing at an exponential rate, and more platforms are needed to accommodate the necessities of a burgeoning list of problem domains. [23] I also realize that resources may be scarce when a developer is confronted with a constrained set of programming languages or proprietary hardware. I think it is prudent that open source software and hardware lends itself to such a do-it-yourself endeavor as this. There are countless forums and articles that abound with information on how to interface one component with another, but rarely do you see a complete system designed in such a way.

For this very reason I have chosen electronic hardware that is supported by twenty different languages. After extensive research, no other robot could be found that is available to the consumer that can say this. Since one of the most important aspects of developing a robot is the computer programming involved, I felt it necessary to not inhibit a developer by creating artificial barriers. To further

this ideal of open source compatibility, I developed a control structure and user interface using open source web technologies such as PHP for scripting, MySQL for data warehousing, and Java for my development environment. This is not to say that one must use the structure I created, it merely gives a jumping off point for the developer. If the user can have a push in the right direction, it reduces the already steep learning curve of robotic development.

### **2.5.1. Market Analysis**

The robotics market is mainly relegated to home cleaning robots, security and military applications, and industrial manufacturing with a small area left for entertainment robots. The hobby industry is growing at a steady rate, so much so, that the de facto robot magazine, aptly named *Robot Magazine*, has already increased its issues per year from four to six. While this is not a stalwart indicator, it does show how this area of interest is on the rise. From reading many articles and posts across the Internet I have surmised that consumers are not able to build a substantial robot to meet their needs, but also cannot afford currently full-featured robot platforms. This puts Axiom Robotics in a great position to offer a user-friendly low-cost alternative.

### **2.5.2. Available Products**

To help robot hobbyists, there are some crossover products that help bridge the technology divide. The radio control car industry has many motors and rotational actuators that can easily and cheaply create movement. There are many off-the-shelf microcontrollers that interface seamlessly with these components. Batteries, fasteners, and other mechanical parts are readily

available if you know where to look. These inexpensive parts, when incorporated together, form a strong electro-mechanical base to start from.

The recent advent of USB-enabled, API-enhanced microcontrollers has given many developers powerful low-level tools to interact with electro-mechanical hardware. This has seen the rise of so many interesting displays of technology that a whole section could be devoted to it. Seeing entire communities based around electronics no bigger than a credit card definitely bolsters my confidence in the longevity of such technology. Another useful product that has made robotic development easier is the plethora of camera choices. In essence, if the robot is unable to relay its environment visually, it is quite complicated to be aware of its surroundings. It also increases the user's enjoyment of their robot when they can peer through the robot's "eyes". Web cameras have become very economical and offer some great features that aid in the robot problem domain.

### **2.5.3. Available Resources**

To go along with the aforementioned hardware, there are subsequent programmatic resources that have aided in robot development. While it is true that there are robot-specific environments that can be obtained for free, I felt it necessary to explore other options before making a decision. I knew that tele-operation was a main goal of this project, so I began by looking at what resources were available. It was evident from the beginning that being able to have concurrent data accessible over a network was essential, and the first thought that came to mind was a database. Databases are specifically designed

to be fast, accurate, and accessible over a network. MySQL was a natural choice due to its speed, small footprint, and most importantly its cost, or lack thereof since it is free.

Since running this system over a network was paramount, I delved into other web technologies to see what resources were available. The Java programming language was an easy choice since Java runs in a web browser with ease, and since it is open source. I was able to utilize Java on both the user and robot sides for control and information purposes. PHP was another web language that was an easy choice, since I used this language as a stepping-stone to create an interface mockup.

### **3. Competition**

Previously mentioned was the scarcity of available platforms for developing robotic unmanned ground vehicles. There are over a dozen robot development companies that might be considered as competitors, but I would like to make a distinction between as either being direct or indirect. This delineation is based on whether the robotic platform has an on-board computer or not. A direct competitor would thusly be a robotic platform that carries an on-board computer similarly to the computer on my prototype robot. Indirect competitors on the other hand are motorized chassis that are controlled via radio control. This distinction is evident in that the non-computerized chassis are neither capable of processing any information from its environment nor is it able make any decisions regarding locomotion. The direct competitors number ten in

all, and in the following sections I will explain the critical differences between my robot and theirs.

### **3.1. Competitor Analysis**

Since it would be prohibitively expensive to be able to test out each and every robot, I am using available specifications for making comparisons. Some features are unique to my robot, others are universal across the whole set. Collectively, the robots vary in size, methods of construction, as well as number of sensors and components. Before making a comparison between these robots it is important for us to look at the companies that produce them. As varied as the robots are, so are the companies. Some companies are publicly traded and have annual revenues in the millions, while others are private and have less than five employees.

#### **3.1.1. Company Comparisons**

**Appendix C: the Companies spreadsheet shows information regarding each company that builds a competitive robot.**

Coroware is a publicly traded company based in Redmond, Washington. With Microsoft's headquarters being directly across the street, it is no surprise that the majority of the 36 employees on staff are from Microsoft. Started in 2003, Coroware began as a software vendor and tele-presence solution provider and has expanded into selling robotic platforms. Now their business has incorporated robotic platform development into their product line, and have made their robots tailored to the researcher and developer market. They currently have two robots

offered, the first one was the Corobot, and the second is the recently released Explorer.

SuperDroid Robots is a privately held robot platform developer in Fuquay-Varina, North Carolina, who specialize in building custom robots that are designed for surveillance and security as well as tele-presence implementations. They have a plethora of vehicles, but the majority of them are not competitors since they do not have on-board computers. They do have a network enabled tele-presence robot that would be considered a competitor based on some similarities between their product, the RP2W, and the Gigerbot.

White Box is a privately held company in Youngwood, Pennsylvania who specializes in robots platforms for the home. They pride themselves on using COTS industry-standard parts to build their robots. Their Nine series of robots have on-board computers similar to a powerful desktop pc, and they also offer different programming constructs, which makes their platform quite inviting. Also, their Nine series of robots have a distinctive cylinder shape to them that visually separate them from the rest of the products compared.

WowWee is a privately held company in Honk Kong, China and they are in a slightly different vein as the previous companies. WowWee specializes in toy robots that surprisingly have some advanced features found in more expensive and more industrial platforms. One of their robots has four legs that can walk and interact with other robots while their “Wrex The Dawg” robot mimics the look and personality of a dog. They also make interactive stuffed animals, aerial vehicles, and bipedal robots. One of their products, the Rovio, is a semi-autonomous robot

that can be controlled using a web application, which is the only offering that would be considered as a competitor. They employ around 125 employees and have been in business much longer than any of the other companies.

The Surveyor Corporation is a privately held company in San Luis Obispo, California that focuses their products on small robotic air and ground vehicles that use their proprietary cameras. Their SRV-1 robot was being tested for the 2012 Google Lunar X Prize competition. According to their website, "The Google Lunar X PRIZE is a \$30 million competition for the first privately funded team to send a robot to the moon, travel 500 meters and transmit video, images and data back to the Earth." [24] The SRV-1 platform is quite small but does incorporate powerful components to make it a strong competitor.

Dr. Robot is a company in Markham, Ontario, Canada and was started by a group of scientists and academics that came together to build cutting-edge robots. By using proprietary hardware and software this privately held company has been able to develop five series of highly configurable utility robots. Their system is also one of the few that allows for tele-presence operation. They make three robots that will be compared: DRK, X80, and I90. Each one has its own unique characteristics and price points that help frame the target market.

iRobot is the publicly held manufacturer of the Roomba robotic vacuum and currently has headquarters in Bedford, Massachusetts. iRobot employs the most employees at 418 and has annual revenue of \$189 M. The company began when an MIT professor started it with two students, and now it is one of the leading robotics companies in the world. Their robots are found in more homes in

the world than any other. I am including a few more companies that are larger more industrial robotics companies since iRobot considers them to be competition. [17] These companies are Mobile Robots, RoboSoft, and Evolution Robotics. Mobile Robots is a privately held international robotic platform developer with corporate as well as military customers. Their extensive onsite development facility in Manchester, Massachusetts has allowed them to create revolutionary software. Robosoft is a company that makes industrial robots in Bidart, France, and is considered in this section since it makes small UGVs much like the iRobot varieties. The Wifibot4G is a competitor since it comes equipped with an on-board computer and its open-source development platform. Finally, Evolution Robotics, a privately run Pasadena, California company that makes robotic subsystems such as mapping, localization, and image processing hardware and software packages is also being considered.

### **3.1.2. Product Comparisons**

**Appendix C: the Robots spreadsheet gives a detailed comparison of salient robot features.**

In the following section, the features of the Gigerbot will be compared with the offerings from other companies. Numerous features will be compared consisting of mechanical hardware, electronic hardware, built-in functionality, software flexibility, and price. There are obviously some companies' products that wouldn't be a good comparison due to the high cost of the hardware. For instance, iRobot who is one of the leading robotics development companies has a plethora of UGVs available, but the lowest cost one they offer starts at

\$55,000.00. It is in the best interest of the following comparisons to limit the scope of the robots compared to ones that cost less than \$10,000.00 since many of those robots have similar size, features, and style of construction as to that of the Gigerbot.

The Gigerbot has a 1.6 GHz processor computer complete with a 12 Gb solid-state hard drive, and 1Gb of ram. The hard drive is the largest of all of the competitors that utilize solid-state drives. Five infrared distance sensors help navigate the Gigerbot which is above average compared to the competitors. Treads are used for transferring the power from the motors into locomotion, and this feature is only shared by two other robots in the comparison. As mentioned previously, the Gigerbot is open source and many of the other robots share this feature as well, but only one other is compatible with the four main operating systems in use today. What is significant regarding the development environments is that the Gigerbot supports twenty development languages, which is far more languages than any other robot in the comparison. Much like the other robots, my robot utilizes a higher-level framework language. Another uncommon feature in the Axiom Robotics offering is a laser pointer, which is quite useful for pinpoint objects with a video camera. The Gigerbot unfortunately has one of the smallest capacity batteries and therefore has the shortest operating time, but this is easily remedied with a larger capacity battery. The majority of the robots including the Gigerbot have some sort of camera but roughly half have web enabled IP cameras. The Gigerbot's IP camera is actually the same unit used by the SuperDroid unit. Two-way audio, high-speed 802.11n

wireless, collision detection, night vision, and video recording capabilities are what helps set the Axiom Robotics' unit apart from the others. Autonomous operation, tele-operation, and web control also help set apart the Gigerbot from the rest since only one other robot supports all three of those features.

The Corobot, the first offering from Coroware, is our first robot to compare. Much like the Gigerbot, the Corobot has an on-board 1.5 GHz computer with 1Gb of ram and a 80Gb hard drive that is comparable to a desktop machine. Tires are used for locomotion and there are two infrared sensors used for obstacle detection. Their development environment is intended for Microsoft languages on the Windows version of the robot, or any language in Linux, if the Linux model is chosen. Since Coroware is an offshoot of Microsoft, it is only natural that the framework language of choice is the Microsoft Robotics Studio. Notable features are the large 80Gb hard drive, the large capacity battery, and the ability to use a joystick controller. Coroware's second robot, the Explorer is quite similar in hardware specs to the Corobot, but the processor is 2.0 GHz, the fastest in the competition, one of the largest capacity battery, the fastest wireless network protocol, and the only robot that can boast having GPS. It is also worth noting that many of the features of the Coroware robots are only in the Windows version of the robots. The Corobot ranges from \$2,874.00 to \$10,404.00 while the higher priced Explorer ranges from \$7,499.00 to \$12,999.00. The Explorer is the highest priced robot in the comparison.

SuperDroid's RP2W is a radically different robot from the rest of the competition with it being the tallest and largely featureless unit. One of the main

reasons for comparing this unit is the fact that the Gigerbot uses the same IP camera as the RP2W and the fact that SuperDroid is headquartered in native North Carolina. The RP2W is the only robot to have a screen mounted on the robot to allow for tele-presence and tele-conferencing. The height of the unit is useful in bringing the person on the other end of the tele-presence connection to eye level as if you were talking to them directly. The lack of an on-board computer severely limits the unit from expansion and programmability, and the only language offered to interface with it is Microsoft VB 6.0, which is an ancient language in today's standards. Also notable is the fact that this robot is the heaviest at 81 pounds and also has the largest battery life at 8 hours. The RP2W costs \$6,940.00.

914 is offering the PC-Bot, an interestingly shaped robot that resembles the shape of the *Star Wars* R2D2 android character, with its cylinder shape. It also sports a powerful 2Ghz processor that is housed in a familiar ATX style case that is the core of the chassis. Injection molded body panels give it its distinctive shape. One of the best features of this robot is the fact that it accepts industry standard computer parts in its computer parts bay. Its price ranges from \$4,995.00 to \$7,995.00.

WowWee's Rovio is quite an impressive little platform that is not only feature rich, but also the cheapest unit compared. At \$289.99 this unit has streaming audio and video, tele-operation as well as autonomous control, and infrared sensors. The Rovio is also the only unit with omni-directional wheels, rather than tires or tracks, which due to the nature of omni-directional wheels,

outdoor use is limited. What seems to prevent this unit from outperforming all of the other robots is the fact that this unit is more toy-like with its all plastic construction, and lack of a programming interface other than an HTTP API, and adding your own hardware is unfortunately not possible.

Surveyor has an interesting and quite small robot called the SRV-1Q. Recently this robot was being tested to be used as a moon rover for the upcoming Google X-Prize competition. It has tele-operated control as well as autonomous functionality much like the Gigerbot, as well as having not one, but two lasers on board. It's proprietary onboard Blackfin BF537 processor can use an interpreted C programming language, or can be console controlled using various languages. The robot can even be controlled via a web application and record video much like the Axiom Robotics robot. It has multiple frameworks that it interfaces with such as the Microsoft Robotics Studio. With a price of \$475.00 it is a tough contender for the Gigerbot to beat.

Dr Robot has a few robotic vehicles that they offer, and I will compare three of them. All three robots use a wheeled type of locomotion, and share similar construction methods. The main differences among all of them come down to variations in shape and electronic components. One of the most attractive features of all of these robots is that they offer vision-landmark navigation that purports to allow the robot to navigate autonomously. This coupled with an auto-recharging feature where the robot seeks out a charging station makes this a unique offering compared to the other robots mentioned. The first is the DRK8080, the only robot that has an animated face and dual

cameras. By using a wireless connection, a user's computer is used to run software that controls and interacts with the hardware on the robot. The DRK8080 only has 1 MB of memory and a small 120 MHz MIPS digital signal processor and integrated microcontroller. This robot does not have an on-board computer as many of the previous robots have had. Since the wirelessly linked PC does most of the processing, less hardware is necessary on the robot itself. It is open source, but only supports Visual Basic and Visual C++, which limits users. It also has a small resolution camera, but by using the available software, one can perform vision processing. The only architecture available for this unit is a proprietary software package developed by Dr. Robot. The DRK8080 sells for \$2,958.33. It is interesting to note that the construction of the Dr. Robot vehicles mimics that of the Gigerbot in that there are multiple plates that are screwed together with spacers in between. This is a predominant theme in many of the robots that are being compared. The Dr. Robot X80 has many of the same components and features that the DRK8080 has, such as the plate and spacer style construction, and on board electronics. The X80 has a slower 40 MHz MIPS digital signal processor microcontroller, and even less available ram, which totals 36Kb. The languages supported are the same as the DRK8080 as well. One great feature of the X80 overall, and in comparison to the DRK8080 is that the X80 has seven infrared distance sensors and three sonar sensors. In all, the ten sensors on the robot total more than any other robot compared. The X80 sells for \$2,587.96. The third offering from Dr. Robot is the I90, and like the other offerings from this company, it is constructed in a plate and spacer form. The

electronics are identical to the X80 in both the amount and kind of sensors as well as on board digital signal processor and microcontroller. One of the nicer features of the I90 is the addition of a high quality pan-tilt-zoom camera that surpasses the cameras on the previous two Dr. Robot vehicles. With the highest resolution and the incorporation higher frame rates, digital and optical zooming features and two way audio, it is understandable that this unit sells for a higher price tag of \$5,555.55.

The last robot I will compare is the Wifibot 4G from Robosoft. Similarly to the Dr. Robot robots, the Wifibot 4G has a 400MHz MIPS processor with 64MB of ram and 32MB of storage. This vehicle uses tires like many of the other robots and it does include two infrared distance sensors as well for range finding. The electronics are geared towards using a wireless network to connect to the microcontroller as well as to operate the pan/tilt IP camera. The robot can be controlled only using C++ and it designed to be used with the RTMAPS C++ robotic framework from Intempora. While this software appears to be flexible and powerful, this might not be so appealing to developers that prefer a different language. Through this programming interface, tele-operation and autonomous behavior is achievable. The cost of one of these robots is \$4,086.09.

There are many robotic platforms available, as we have seen with varying features that make it a tough choice on which platform is best. It is ultimately up to the developer or hobbyist which platform would best suit their needs. This seems to side with the notion that robots are still be purpose built, and not having the flexibility necessary to accommodate a varying user base. This shows that

there is room for the Gigerbot that in many respects has a greater set of overall features than any of the other robots compared. In the following section I will show how Axiom Robotics can provide a superior product at a competitive price.

### **3.2. Opportunity Analysis**

Since it is such a complex task of building a robot, companies have found a market for developing platforms to appeal to the mass of developers, researchers, and hobbyists. From the previous comparisons, it is obvious that the median level of technology seen on the robots includes a locomotion platform, an on-board processor, wireless connectivity, a vision system, a sensor system, and some type of programmable hardware. There is a lot of variation in the size of the robots as well as the costs. A competing robot would need to fall in line with the group of small UGVs that have autonomous, tele-operation, or both. In the following section I will delineate a low-cost open source hardware and software platform that not only has outstanding features but also has an attractive price. It will be demonstrated how the Axiom Robotics UGV is superior in overall value and richness of features

## **4. Methodology and Plan**

In this section I will explain how a low-cost open source robotic platform can be designed with flexibility in mind for use by developers, hobbyists, and researchers. Using off-the-shelf mechanical hardware, multi-language software supported electronic hardware combined with a database-centric application framework will be shown to be a formidable product in comparison to the

aforementioned platforms currently available. Superior hardware, software, and technology combined with an attractive price tag will be shown to be the best platform for the small UGV market, and this will demonstrate the strength of this product.

In the summer of 2009, Axiom Robotics received first place in a robotic competition. Lynxmotion, a supplier of robotic platforms and parts, had an open competition for robots built from their parts. There were fourteen contestants, and judging was based upon complexity, uniqueness, creativity, functionality, and presentation. The competition was comprised of university professors, college students, and industry professionals all vying for the top position. By showing how the Axiom Robotics prototype is not only functional, but also geared for developers, it was clear my robot was the superior model for flexibility and cost. This only bolsters my confidence that I have developed a solid platform that consumers will be eager to buy and use in their further development.

#### **4.1. Product Description**

There are many attributes and desired features that go into developing a robotics platform, and therefore it is appropriate to discuss the development of this product beginning with the most basic attributes and building from there. In essence, I will describe the building of the robot in same order that I tackled each hurdle. A ground up approach was taken in deciding components and functionality, and it is my fervent belief that strong design principles facilitate more robust end products.

An overview of necessary functionality is the ability to move not only the vehicle but also to be able to position a camera for vision purposes. The robot would need to sense its environment and be able to provide some feedback as well. The control of the robot would need to be done wirelessly, preferably over a network. The robot must also be able to function on its own or be able to be controlled by a user. These choices as well as other design aspects were taken into account to further define the prototype.

#### **4.1.1. Technology**

A small UGV would of course be quite useless if it did not have some sort of locomotion, as in a way for the vehicle to move, and to orient any sensors into position. To tackle rough terrain, designers of military vehicles have turned to treads for weight distribution and effectiveness in muddy terrain. After much searching, I came upon a small, lightweight tread system created by Lynxmotion. I purchased a simple kit, which included the treads, motors, and chassis plates necessary to have a simple treaded platform. At this point the robot was not moving, but I had a way to move. I also realized that I would need some sort of camera so that the robot would know where it was going. A pan/tilt unit was sourced from a company called ServoCity, and I was able to purchase servo actuators, which provide rotational movement when interfaced with an RC style control system. Essentially, the servos would move according to Pulse Width Modulation, where the desired rotation was expressed in timing signals inputted into the servo via a small wire. This is an industry standard in the RC world for hobby airplanes, cars, boats, etc. At this point I had all of the pieces necessary to

make the robot move, but I had no way of telling the hardware how much electricity to put into the devices. I still had a paperweight.

Fortunately, the RC industry also makes “radio” systems, which are glorified transmitters and receivers specifically designed for RC vehicles. They are characterized by simple FM frequency transmission and reception, and this style of control is used by many robot builders to make sure that the locomotion systems work, as they should. However, the use of a direct wireless control of the robot via RC systems does not fit the definition of a robot. I like Wikipedia’s definition of a robot as a “...virtual or mechanical artificial agent. In practice, it is usually an electro-mechanical machine, which is guided by computer or electronic programming, and is thus able to do tasks on its own. Another common characteristic is that by its appearance or movements, a robot often conveys a sense that it has intent or agency of its own.” To this end, if the robot is only responding to the commands given over airwaves with no consideration for sensors and outside aspects and having no computer controlled functions would not qualify as a robot. I need to find a way to manipulate the motors and servos with software. [25]

There are a plethora of devices available today that can be connected to a computer with a full API and software implementations. After searching for devices with the right mix of functionality, price, and support, I found the right components. Phidgets, a Calgary, Alberta, Canada microcontroller provider has hardware that more than met my needs. The Phidgets family of electronics has a strong programming backbone that includes twenty programming languages that

are part of the API. I knew that my development language needs might change over the course of this project, and I also knew that my client base might not have the same programming skills or available software development environments necessary to work with my system. By choosing such a wide-open microcontroller choice, I knew that this would ultimately be a strong selling point.

**Appendix A: Figure 6 shows the Phidgets microcontrollers.**

I ended up purchasing three devices: a servo microcontroller, a motor microcontrollers, and a sensor with digital switches microcontroller. The servo microcontroller allows the software to specify the rotation amount and velocity of movement of the servos that maintain the position of the camera. Next, the motor microcontroller allows me to set the speed and direction of the treads in an effort to provide a smooth skid-steering control. [25] Through software, I can drive the vehicle in any direction and simultaneously move the camera to any position necessary. The third device, the sensor and digital switch microcontroller interfaces with infrared distance sensors and are able to return a digital value indicating the distance of an object from the sensor. This is coupled with software-controlled functionality that will close or open a digital switch to turn on lights, lasers, and other components. Now, not only can the robot move, but also it can sense its environment.

**Appendix A: Figure 1 shows the Java Agent diagram.**

It is important to note that one of the chief aspects of the robot's functionality is the ability to switch from the AI control of the robot to the user control of it, and back again. This presented a problem whereas the telemetry of

the vehicle must be concurrent and uniform between the AI and the robot when in autonomous mode, and between the user and the robot when in direct control mode. To further compound the issue, the user and robot must be able to be in separate locations across a network. The pertinent telemetry data must be portable across the network, it must be fast, and it must be able to handle multiple concurrent connections while ensuring data integrity. The frequency of reading and writing of these values meant that mechanisms must be in place to prevent data corruption.

**Appendix B: shows the database structure.**

After much deliberation I decided to use a MySQL database to house the telemetry data. I had considered writing a standalone data warehouse, but this proved to be too complex as well as not developer friendly. I also considered complex message passing, but this too fell short since data concurrency wouldn't be preserved. The MySQL database had readily available data connections for a myriad of languages, and this along with its open source availability was a strong selling point for going with a database option. At the time of this writing, there are eleven variables that must be maintained. Five of these values are distance sensor values, two are turret direction values, and the remaining four are relegated to the treads. The reasons for choosing a database are based on the fact that databases can be used across a network with great speed, they can handle multiple concurrent connects, they have built in optimization, and they are flexible when modifications are necessary. MySQL is also easy to use, and since it is open source, it is subsequently free.

**Appendix A: Figure 4 shows the server side web interaction.**

To implement this structure, the on-board computer acted as a web server with MySQL and this therefore allowed me to have local access to the database. Locating the MySQL web server on the on-board computer as opposed to an off-device computer was chosen since the majority of queries were from the driving software on the robot. Now that we have a data scheme, it is necessary to connect the database to the robot's microcontrollers, and to do this I chose Java. Java is open source, is free, and has a strong developer community. Its breadth of capabilities made it an easy choice to guide the robot. There are two main tasks that the software of the robot executes. First, the program updates the database with the sensor values as well as the movement calculations made by the built-in AI. The program must also adjust the motors and servos according to the values in the database. This two-part sequence is essential to keeping the robot and end user synchronized. To achieve any level of controllability, the aforementioned sequence must happen very rapidly, and I found that a twenty-millisecond delay between iterations provided a comfortable delay for control.

Another server residing on the robot is the IP camera sitting on the turret. This unit provided by Zonet is a full-fledged web server that has an API and a hackable interface. I disassembled the web-based video player, and retrofitted the video Java Applet into a web page served by the on-board web server. This allowed me to have a graphical user interface using HTML, CSS, and images. Once again I am using open source languages that readily available tools can be found to manipulate. By using PHP, an open source web programming language,

I was able to use the web server to respond to web page clicks to manipulate the database residing on the robot. This gave me a primitive graphic user interface with view and control aspects. To further develop the interface, I added a Java applet dashboard that polls the database every fifty milliseconds for the status of all of the values in the database. This lets the user know at what speeds and orientations the components are at, as well as the values of the five distance sensors.

To enhance the control of the robot, it is my goal to interface a gamepad with the robot in an effort to make the tele-operation of the robot very similar to a First Person Shooter video game. [26] This has been one of the most painstaking processes since there is sparse support for gamepads and web based gamepad APIs, and I currently do not have a solution to the problem. Hopefully, this will be resolved quickly, and a true test of the interactivity of the system can be achieved. Also, as the last few hardware additions are being decided, I have acquired a laser pointer, which will give targeting and depth references. The laser pointer is a necessary addition, which will add very little to the cost of the robot.

In summation, I have successfully navigated the robot without visibly seeing the robot and navigating solely with the robot's camera and inputted sensor data. The farthest distances achieved were around one hundred feet, and by that point the thick walls were impeding the needed signal strength that the robot needed. In a more open environment, I am assured the distance could be lengthened.

### 4.1.2 Business

As we have seen in previous sections, the robot I have built compares with many of the features of the competitors that are priced from \$300 to \$13,000. The costs for the current prototype is around \$1,500, which is a considerable savings given that it has an on-board computer, out of the box AI and user control, and is easily controlled over a network. The multi-language open source approach leaves a lot of options available for the researcher, developer, or hobbyist alike. If the developer does not like the architecture, then the APIs are available for any part of the software that needs to be rewritten. By offering a robot that is not only usable from the beginning, but is also customizable on many different levels, it makes for a fertile development environment.

One of the great aspects of the business side of this prototype is that the majority of the parts can be ordered and shipped directly to me, to incorporate into the robot. There would be very little overhead for purchasing tools, materials, and other things that drive up not only sunk costs, but also operating costs. The operations of advertising, sales, and support can be accomplished through a corporate website, which is yet again a relatively low operating cost. By keeping costs and overhead low, and by reducing the amount of time necessary to build the robot product, it will allow me to be competitive in not only pricing, but also product availability. Some robot manufacturers require four to six weeks for delivery since their robots are built to order, yet I can easily build a robot in one day.

### **4.1.3. Product Uniqueness**

The use of a database is not a structure that I have seen used by other robot developers. While I do not definitively know if any other developer has used a database in the manner that I have, I do know that it is at least rare. I also know that there are very few robots that also have AI and tele-operation out of the box, and with these two features coupled, I would venture to say that I have a unique product. This separation is pertinent to the success of the robot as well as to Axiom Robotics.

### **4.2. Strategy and Operations**

In this section I will be further describing how the product will be produced and sold to complete the total overview of how Axiom Robotics will operate and be successful. I will delineate how this venture will create value and how it will react to the market environment.

To begin this section it is important to analyze market trends in the near term as well as long term. The United Nations Economic Commission (UNEC) and the International Federation of Robotics (IFR) both agree that the personal and service robotics market is booming. Between 2002 and 2005 the robotics industry doubled, and had reached \$5.2B by 2005. By 2010 the industry will exceed \$17B. It is estimated that this market will increase to over \$50B by 2025. The UNEC went on further to show that in 2003 the largest surge in robotics was in the United States with a 28% gain. Another point came from an October 2004 press release from the UNEC stating, "Profitability studies have shown that it is not unusual for robots to have a pay-back period as short as 1-2 years." This

would seem to bolster the fact that robots are becoming a more profitable venture and many individuals will need robotic development platforms to help them realize their goals. In the period of 2004-2007, professional robots have increased more than double of their 2003 numbers. As an ancillary residual effect from the build up of military robotic technology “[h]uge military investment in service robots will give spin-off effects both for the market of professional service robots and for the market of consumer products.” It is easy to see that there will be exponential growth in the robotics industry in the coming years, and Axiom Robotics is poised to seize this opportunity. [32, 33]

The product that Axiom Robotics will be selling is the Gigerbot, a fully functioning and expandable robotics development platform. The business model is two fold. First I want to sell these customizable robots to consumers for their own development and enjoyment. Secondly, once these customers have my product, this will leave give me an avenue to sell upgrade components that will further bolster sales. This customizability will eventually help the product expand as more popular upgrades will help define the usability of the Gigerbot and hopefully create different versions of the product for sale. This is a tenant of the fundamentals of how this product is being created. By developing a powerful generic platform, more specific platforms can be derived from it. For instance, one of the first upgrades I have planned is a robotic arm that can be easily attached to the Gigerbot to allow for manipulation of objects in the robots environment.

It is also important to note that I will also create a community based forum that will work closely with current and future clients to help them decide whether Axiom Robotics robots are right for them, and also see what market trends are occurring. By staying in close touch with our customers, it will make it easier to create products that already have an eager group of potential buyers awaiting its release. It is absolutely vital that a strong core group of users be created and sustained to keep the product line vibrant.

Axiom Robotics plans on selling the Gigerbot in two different forms. The first would be a turnkey robot that comes pre-assembled and pre-loaded with all software. The customer would only need to charge the batteries, turn the unit on, and perform some minor configurations. The Gigerbot would be able to come alive, and navigate using its multiple operating modes. The second form would be a kit, which comes complete with all of the components and software necessary to assemble a Gigerbot and would function no differently from the turnkey unit. The only difference between these two forms of the Gigerbot is the price. The customers would be able to save money by choosing to assemble the robot themselves, and I feel that many of the potential customers would want to take the time to assemble the robot themselves. This would give a greater understanding to the user on how the robot operated and also on how they could upgrade their Gigerbot to fit their specific needs.

There will be one route to market initially, but over time I would like to have two different routes to market. It is my understanding that operating a sales department can be quite costly, and to reduce the expenses of selling my

product, I have opted to use an online store to initially sell my products. The cost of setting up and maintaining an online store is substantially cheaper as we will see in the financial section of this paper. Eventually, I would like to wholesale the Gigerbot to other robot online stores to bolster visibility and to increase sales. Many of the robots that I have compared in previous sections only sell to the public via their own online store, but only a few sell to other retailers. This dual route to market should increase Axiom Robotics potential stronghold in the target market.

From Appendix C the mean and median competitor prices are \$3,328.00 and \$3,522.00 respectively, and by comparing the cost of goods sold for the Gigerbot at \$1,848.13 from the financial section you will see that there is a margin of 55.6% and 52.5% respectively. This is a healthy margin to have, and this margin can further be widened through the reduction of costs through sourcing even more cost effective parts.

Advertising for Axiom Robotics will consist of two different types of media: print and online. There are two magazines that I have read for years and seem to have a wide following of diverse readers. The first is aptly named *Robot* and the second is *Servo Magazine*. By advertising with *Servo Magazine* alone, this would garner 42,000 impressions per month, and advertising with *Robot* magazine would put Axiom Robotics in front of another 35,000 individuals. One of the perks of advertising with *Servo Magazine* is that when you advertise with the print version of the magazine, they also put your ad on their website. With both magazines being focused on the target market, it is assumed that the conversion

rate would be higher than with other outlets of advertising. I also plan on using email-marketing strategies to let current and potential customers about new products. Since I am familiar with email marketing, and since it is a cost free endeavor, this will also increase my impressions, without impacting the bottom line. [31]

The other form of marketing would be through search engine optimization (SEO). As mentioned previously Axiom Robotics' website, online store, and forum would be the main points of contact for potential and current customers and it would only be natural to leverage these web presences through SEO. There are two forms of SEO, the paid type, and the unpaid type. If one pays for search engine placement, then that is considered pay-per-click advertising, but if one manipulates the content of their site in order to increase its ranking which is at no cost to the owner, then this is considered organic SEO. This free ranking manipulation of course takes more effort, but the cost is substantially cheaper than pay-per-click. This organic SEO would be an easy way to help bolster incoming traffic that would substantially increase Axiom Robotics visibility in the marketplace.

There would be four main operations in the company: research and development, sales, production, and management. Initially I would perform all three operations but as the company grows, there would need to be employees to fill these expanding roles. Sales would be handled via the Axiom Robotics online store and after that, the production department would begin work on preparing the product for shipment. It is estimated that it would take sixteen man-

hours to complete one turnkey robot, or one hour to prepare a robot kit for shipping. Since the majority of the parts are COTS or custom made at a third party facility, very little work would need to be done in house except for some wiring harnesses that would need to be made. This also reduces the need for a large space or expensive equipment to assemble a robot. A sale would be complete once the production department has shipped the unit off to the customer, and this entire sales and production turnaround time would take roughly two days. Research and development, which would be headed up by me, and would serve to improve upon current products and processes as well as develop new products. Lastly, the management of the company would of course need to encompass the financial, personnel, and general management of the company.

The Gigerbot is obviously a superior product, and once it comes to market other companies may see the successful implementation of newer technologies and may start to mimic these improvements in their own product. This move may initially shorten the technology gap between Axiom Robotics and the competition, but by being attentive the needs of my customers and incorporating newer technologies on a regular basis this should sustain Axiom Robotics position as a robotic technology leader.

To date Axiom Robotics has accomplished some significant milestones on the road to a successful company. By already having a working prototype, this solidifies the product line as well as helps lay the framework for a solid venture capital plan. With the conclusion of this document, a business plan will be

complete, and will allow me to be ready to acquire funding. Axiom Robotics also already has a web presence with its website axiomrobotics.com. There are however a few more milestones that would need to be completed before the company would be ready to begin operations. I would need to finish the company website which would include setting up an online store, and I would also need to solidify who would be manufacturing my custom plates. After that I could focus on acquiring a small space to run the company and finish my packaging and product materials such as a manual. Once these milestones were complete, Axiom Robotics would be ready to procure a business license and open its doors for business.

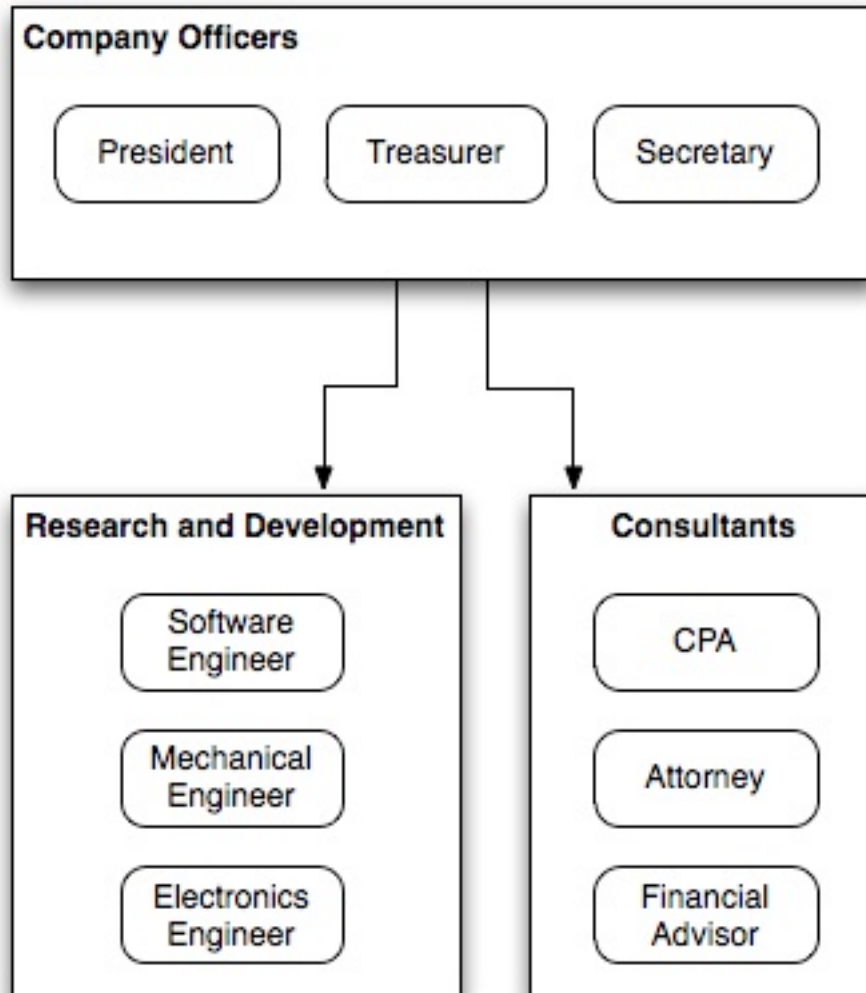
### **4.3. Management**

In order to define a management scheme for the company, it is necessary to define what type of corporate structure Axiom Robotics will have. After much deliberation, I think the best route would be to set up a limited liability corporation for the first two years, then convert the company into a subchapter c corporation from year three and on. The benefits of this, is that the LLC has fewer yearly taxes, less legal obligations, and much less paperwork necessary to keep the company running. This would give Axiom Robotics time to develop its core market share, and strengthen as a company. Referring to the gross income by year three, this would give the company a stable operating income to sustain the rigors of a c-corporation. At this point Axiom Robotics could begin to seek out venture capitalists and angel investors to facilitate for expansion. Another important aspect of having a c-corporation is the ability to issue shares of stock. I

think it is quite important that all employees are invested in the venture, and by offering stock options to all employees; it would give an incentive to them to not only stay with the company, but also work harder for the greater glory of Axiom Robotics. [31]

To manage Axiom Robotics initially I will bear the brunt of each and every department. As the company grows, there will be three distinct groups of individuals that will be needed to run the company. To run the corporation I will need officers to serve in the capacities needed to file required quarterly and yearly documents with the state and federal government. These officers will need to interact with the second group of individuals that will function as consultants. The last group needed is the employees that develop the products that Axiom Robotics will sell. By having all three groups working together, we will have all the necessary ingredients to develop, market, sell a product and be able to make a profit in doing so.

## Company Organization Chart



The corporation will need a handful of officers to facilitate the necessary running of the company. I will be the president and at first I will also be the treasurer and secretary. It is advisable that the treasurer and secretary positions should be held by someone other than the president. I plan to nominate my father, who is a financial advisor to the treasury position, since he is has a wealth

of knowledge on the subject matter. At the right time, a secretary will be found to fill that position. As a corporation I do not see a need to include any other officers except the bare minimum for corporate governance.

The company will also need consultants from time to time to make sure that we are not only in compliance with state and federal laws, but to also protect our intellectual property. An attorney would need to be retained to make sure our corporation stays valid, and to begin the process of acquiring trademarks and patents for Axiom Robotics' products, services, and concepts. To help keep the financial books in order, it will be necessary to have an accountant that can prepare taxes for the government. The accountant would also need to liaise with a third consultant, a financial advisor. It is very important to secure investment capital in the short term to have the necessary cash flow for purchasing inventory, paying employees, and paying bills.

On the other hand, to develop the product side of the company, it will be necessary to find engineers for electronics, mechanical systems, and software. Over the first five years I will be securing talented individuals to fill these positions to keep Axiom Robotics at the forefront of technology. I envision needing three full time employees other than myself. The first employee would be needed for developing the circuit boards that interface with sensors, control motors and servos, and toggle devices. The second employee would need to be a software engineer that would help with developing software to interact with the electronic hardware, and to help design the overall architecture of the system. The third employee would need to be a mechanical engineer and he would be

instrumental in developing the chassis, tread assemblies, turrets, and arm assemblies for future revisions of the Gigerbot.

#### **4.4. Financials**

Appendix G shows the estimated quarterly sales, cost of goods sold, advertising costs, operating costs, and quarterly budget. Throughout this section I will delineate how careful planning and considerations are given for the financial management of the company. It will also show that Axiom Robotics will be profitable within the first year of operations, and be well on the way to being a successful c-corporation within three years. [31]

Starting with the cost of goods sold (COGS), there are two products that Axiom Robotics will offer, one being a robot kit and the other being a turnkey robot. Both have the same material cost of \$1,540.44, but each has a labor cost associated with it. The kit only would require one hour of labor, which is attributed to taking the various parts from inventory and packaging them together into a box ready for shipping. The turnkey robot requires sixteen hours of labor to assemble and test the finished robot to make sure that the customer is receiving a working unit. I estimate labor on an hourly basis based on a nominal \$40,000 per year salary. The hourly rate comes out to \$19.23, which would give the final production costs for the kit and turnkey robots at \$1,559.67 and \$1,848.12 respectively. The estimated manufacturer suggested price for the kit would be \$2,700.00 and this would give a gross profit margin of 42.23%. The estimated manufacturer suggested price for the turnkey would be \$3,000.00 and this would give a gross profit margin of 38.40%. Notice that both suggested retail prices are

below the mean average of the competition by over \$300.00. The turnkey would be undercutting the competition on average by 10.9%. [31]

Another cost associated with building the company is advertising. Placing ads with *Servo Magazine* and *Robot* magazine for an entire year would cost \$3,631.20 and \$2,878.98 respectively. With *Servo Magazine* reaching 42,000 people per issue, and since they have twelve issues per year, then that is 504,000 impressions, which would mean that it would cost \$0.01 per impression per year. Similarly, *Robot* magazine reaches 35,000 readers per issue, and they have six issues per year. This would give an additional 210,000 impressions for a \$0.01 per impression per year rate as well. All in all, I can potentially reach 714,000 robotic enthusiasts with my product for \$6,510.18 per year. [31]

To make money, there has to be a way for customers to order my products. The most cost effective way of doing this is by using an online store to promote and sell the Gigerbot. The beauty of such a store is that it is very low cost, it would maintain inventory, and it would handle all processing of credit cards. To do this, Axiom Robotics would need a domain name and hosting service. The domain is \$10.69 per year, and the hosting is \$12.50 per month. There are free, open-source shopping carts available on the Internet that work quite nicely, and I would employ one of these to run my store. The only other things needed to make the web store work would be an SSL certificate for security purposes, which runs \$29.99 per year, and a transaction processing account for actually making the credit card transactions and depositing the money in Axiom Robotics bank account. The cost of the transaction processing is

thirty dollars per month as well as 2.2-2.9% of the total sale per transaction, and \$0.30 per transaction. Overall it would cost \$550.68 per year to run the store excluding the variable transaction processing fees. These associated fees are reflected in the quarterly budget found in Appendix G. [31]

In estimating the quarterly sales for each type of robot sold, I was unable to find any de facto numbers from suppliers as to the number of units that they sell. I ended up using modest assumptions in estimating sales figures. In the first year, I estimated that one robot be sold per month, with twice as many turnkey robots being sold over the kits. This would equal out to three units per quarter for year one. I feel that the market would be more in favor of turnkey robots over the kits, and have reflected this in the quarterly sales. In year two I estimate ten units being sold per quarter. Year three is the year when Axiom Robotics becomes a c-corporation and hopefully has the necessary investors to expand, which would explain the jump to twenty units per quarter being sold. In year four I estimate that we will still be growing, and by then Axiom Robotics should be outputting 32 units per quarter and in year five, 45 units per quarter. Even if these estimates are overzealous, the quarterly budget shows a significant amount of padding in the gross income to allow for this. [31]

It is my desire to run the company from my home during the first two years as an LLC, which will give me a lower overhead so that I can grow the company to the level necessary to branch out. On the third year, when Axiom Robotics will become a c-corporation, I would also like to have a location of business other than my home. After surveying the targeted area of Raleigh, North Carolina, I

have gathered some estimates on costs setting up shop there. A typical warehouse lease for a 1,200 square foot space would run \$800.00 per month. Utilities would run an estimated \$150.00 per month while Internet from Time Warner Cable costs \$79.99 per month. To reduce costs, I planned on using Vonage for the phone system, and this would run \$39.95 a month. There would also be some associated sunk costs for tools, office equipment such as computers and desks, and office furniture that would also need to be purchased to make the new company office viable. I estimate that on year three, to outfit the new office Axiom Robotics would need \$500.00 in tools, \$1,500.00 in office furniture, and \$5,000.00 in office equipment. The office equipment and furniture costs would cover computers, desks, lamps, supplies, fax machine, and other associated office items. [31]

Lastly, the five-year quarterly budget is a spreadsheet of great importance. This document shows all of the expenses in comparison with the expected revenue to calculate the quarterly gross income. It is noteworthy to point out that only the first quarter results in a negative gross income, while the rest of the quarters beginning in year one is profitable. Albeit there is a modest profit being made, it is important to establish a track record to acquire investors for the third year milestone. It appears that during year three, Axiom Robotics could start supporting one full time employee comfortably. As mentioned previously, there would need to be three full time employees beyond myself, and this could easily be attained granted that Axiom Robotics could garner some outside investments. I plan on paying each employee \$40,000 per year to start with, and considering

that the rule of thumb is that an employee costs twice what their salary is per year. This would mean that I would need \$320,000 per year to pay for three engineers and myself. According to Appendix G, this would not be feasible unless Axiom Robotics acquired some outside investments. This gives me more motivation to not only grow the company at a faster rate, but also to learn other disciplines so that it isn't necessary to hire as many individuals. [31]

It is also inevitable that over the years Axiom Robotics will be able to not only develop newer Gigerbot derivatives that are less expensive to make, but through the economy of scale, also reduce the costs in making each unit. This would positively affect the bottom line, and further increase gross income. By using newer and more cost effective hardware, it is quite possible to even further reduce the costs of producing a Gigerbot, and hopefully this savings can be passed onto customers in an effort to further entrench Axiom Robotics as a leader in cost effective robotic development platforms.

## **5. Conclusion**

Robotics is a growing area of research and development not only in academia, but also in professional and hobby arenas. There have been significant barriers to entry caused by high costs, and limited hardware and software. For some time there has been a need for a platform that is cost effective and developer friendly that not only is flexible to use but can be upgraded easily. Axiom Robotics is a company that can provide what the current robotics market needs.

As is demonstrated throughout this paper, it is possible to create a robotics development platform that not only exceeds the benchmarks set by current competition, but also undercuts the average prices set forth by the aforementioned competition. This has been demonstrated by the prototype Gigerbot I have created, and through the development of mechanical hardware, electronic hardware, and software, this robot is a formidable competitor in the target market. It has also been demonstrated that a company, Axiom Robotics, could be created around such a product and could be successful within a few short years of inception. In conclusion, it is evident that Axiom Robotics is a solid venture and has the potential to become a leader in the robotics development market for years to come.

## **6. Future Work**

It has been identified by a panel of experts, my committee, that there are some areas of future work that would need to be considered. These topics will be discussed in this section to further expound upon work that would need to be done to ensure the successful launch of Axiom Robotics.

It was brought to my attention that there are some added expenses that would need to be accounted for, since they will invariably need to be included at various junctures in the business venture. One of the first expenses that would need to be increased is marketing costs. While the search engine optimization and advertising in magazines are a good start, I might be missing a segment of my market by not expanding my Internet marketing strategy. Since my entire development model is designed around an e-commerce strategy, it is particularly

important to focus on the online marketing aspects. To this end, it was suggested that pay-per-click advertising and keyword purchasing take place to boost the visibility of the site. Fortunately, Google, the premier online advertising agent, allows an advertiser to limit the amount of variable costs they can allow. For instance I can put in a cap of five dollars per day, and Google will not show my advertisements to more people than I can afford. This is a feature I was not previously aware of, and this has changed my thinking regarding using pay-per-click advertising. Another strategy used by search engines to establish credibility and higher rankings in its search engine is the amount of years that a domain is registered into the future. If a domain name is registered for the next ten years, then it is assumed that this site will have more longevity, and therefore should be ranked higher than say a domain that is registered for only the next year. This of course would be another expense that would need to be undertaken at the beginning of the company, and this cost would need to be factored into operating expenses. Lastly, there are other sites that I could advertise on, that I did not fully research that could yield high traffic flows. There are a plethora of websites that are geared towards robotic development, and a few of them do offer direct advertising on them. I think through advertising on these smaller scaled, but more targeted sites that I could reach a higher correlation of visits. By using advertisements on websites, I would be able to analyze which sites provide more clicks and could therefore use this to maximize my advertising budget.

One of my identified target markets is the education market. Robots are a popular choice for students and professors alike, and it would be prudent to find

more avenues to demonstrate my product to them. One of the best ways to get in front of many of my audience would be to travel to various robotic conferences, competitions, and expos. Two notable venues, the international RoboGames competitions, and the international Robo Expo are both prime candidates for setting up a booth and demonstrating the capabilities of the Gigerbot. This endeavor would be a positive step in advertising, but it can be quite costly. Traveling expenses, hotels, and conference fees can be expensive, so I would need to carefully choose one or two events per year to maximize my exposure while minimizing my costs.

Another expense that I did not consider in the original draft of this paper was the cost of employees overall. While my one example employee might be paid \$40,000.00 per year, the rule-of-thumb for actual costs of an employee is roughly two times their salary. This takes into account benefits, human resource costs, and government taxes for Medicare, unemployment, and Social Security. I will need to budget accordingly so that I can afford quality employees without their pay suffering. Strong companies are dependent upon dedicated employees at all levels, and it is paramount to keep an employee happy if one is to keep an employee for a long period of time.

The last expense that I haven't fully considered until now is the cost of customer support. Since I will be selling a hardware product that is closely tied to a software product, there is bound to be support issues. Originally I had planned on using the website and online forum to support users, but this can only supply so much knowledge. It would be prudent to have more support avenues such as

a phone number earlier on in the company's inception. There would also be a support cost if any of the hardware malfunctioned or broke, and I have not considered what type of warranty I would offer. Since many of the parts I am using come from other manufacturers, it would be prudent to see what their return policy would be for Axiom Robotics as a third-party vendor. Hopefully I will be able to use their return policies to my advantage.

I was informed that the UNCW Small Business Center offers free marketing plan development for qualifying companies. I wasn't aware that this valuable service was offered, which would definitely bolster my marketing plan without adding anything to my bottom line. I am planning on setting up an appointment to see if my future company would qualify and begin the process. To add to this extra documentation, I was also informed that a yearly analysis of my company would be a welcomed addition as a corollary to my quarterly analysis. This would provide a summary of yearly activities that would help average out my perceived yearly expenses since most of my expenses were identified in the first quarters of each year in my five-year plan. To also add to this extra documentation I neglected to create a cash flow statement. Since a cash flow statement is an indicator of the short-term viability of a company and whether or not it can pay its bills, this would be an important analysis document to have prepared before I launch Axiom Robotics.

Overall, I believe I have a solid plan for the development and launch of a robotics development company, which has been honed through scientific research, academic feedback, thorough analysis, and adherence to business

standards and practices. It is my goal in the next five years to implement this plan, and be able to put these ideas into practice. This would be the ultimate test of how well this plan would succeed in the business world. Only time will tell.

## 7. References

1. Chen, J. Y. C., Barnes, M. J. "Robotics Operator Performance in a Military Multi-Tasking Environment." In proceedings of the 3rd ACM/IEEE international conference on Human robot interaction. (Amsterdam, The Netherlands, March 12-15, 2008 Pages 279-286)
2. Olsen, Jr., D. R., Wood, S. B. "Fan-Out: Measuring Human Control of Multiple Robots." In proceedings of the SIGCHI conference on Human factors in computing. (Vienna, Austria, April 24-29, 2004, Pages: 231 – 238)
3. Parasuraman, R., Sheridan, T.B., Wickens, C.D. "A Model for Types and Levels of Human Interaction with Automation." In proceedings from Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on Volume 30, Issue 3, May 2000 Pages: 286 – 297
4. Koay, K. L., Dautenhahn, K., Woods, S. N., Walters, M. L. "Empirical Results from Using a Comfort Level Device in Human-Robot Interaction Studies" In proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction. (Salt Lake City, Utah, USA, March 2-4 2006 Pages: 194 – 201)
5. Sakamoto, D., Kanda, T., Ono, T., Ishiguro, H., Hagita, N. "Android as a Telecommunication Medium with a Human-like Presence." In proceedings of the ACM/IEEE international conference on Human-robot interaction. (Arlington, Virginia, USA, March 8-11, 2007, Pages: 193 – 200)

6. Livatino, S., Privitera, F. "3D Visualization Technologies for Teleguided Robots." In proceedings of the ACM symposium on Virtual reality software and technology (Limassol, Cyprus, November 1-3, 2006, Pages: 240 – 243)
7. Forlizzi, J., DiSalvo, C. "Service Robots in the Domestic Environment: A study of the Roomba Vacuum in the Home." In proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction. (Salt Lake City, Utah, USA, March 2-4, 2006, Pages: 258 – 265)
8. Nakanishi, H., Murakami, Y., Nogami, D., Ishiguro, H. "Minimum Movement Matters: Impact of Robot-Mounted Cameras on Social Telepresence." In proceedings of the ACM 2008 conference on Computer supported cooperative work. (San Diego, CA, USA, November 8-12, 2008, Pages 303-312)
9. Glas, D. F., Kanda, T., Ishiguro, H., Hagita, N. "Simultaneous Teleoperation of Multiple Social Robots." In proceedings of the 3rd ACM/IEEE international conference on Human robot interaction. (Amsterdam, The Netherlands, March 12-15, 2008, Pages 311-318)
10. Wang, J., Lewis, M. "Human Control for Cooperating Robot Teams." In proceedings of the ACM/IEEE international conference on Human-robot interaction. (Arlington, Virginia, USA, March 8-11, 2007, Pages: 9 – 16)
11. Nishiyama, T., Nomura, J., Nakajima, R., Miyano, T. "A Semi-Autonomous Security Robot Employing Telepresence." In communications of the ACM archive Volume 41, Issue 5es May 1998 Pages: 198 – 205

12. Goza, S. M., Ambrose, R. O., Diftler M. A., Spain, I. M. "Telepresence Control of the NASA/DARPA Robonaut on a Mobility Platform." Proceedings of the SIGCHI conference on Human factors in computing systems. (Vienna, Austria, April 24-29 2004, Pages: 623 – 629)
13. Shiomi, M., Sakamoto, D., Kanda, T., Ishi, C. T., Ishiguro, H., Hagita, N. "A Semi-autonomous Communication Robot – A Field Trial at a Train Station." In proceedings of the 3rd ACM/IEEE international conference on Human robot interaction. (Amsterdam, The Netherlands, March 12-15, 2008, Pages 303-310)
14. Tanev, I., Joachimczak, M., Shimohara, K. "Evolution of Driving Agent, Remotely Operated a Scale Model of a Car with Obstacle Avoidance Capabilities." Proceedings of the 8th annual conference on Genetic and evolutionary computation. (Seattle, Washington, USA, July 8-12, 2006, Pages: 1785 – 1792)
15. Morita, T., Mase, K., Hirano, Y., Kajita, S. "Reciprocal Attentive Communication in Remote Meeting with a Humanoid Robot." In proceedings of the 9th international conference on Multimodal interfaces. (Nagoya, Aichi, Japan, November 12-15, 2007, Pages 228-235)
16. Coroware 10K Annual Report. Securities and Exchange Commission. 15 June 2009 17:51 UTC. 15 June 2009.  
<<http://www.sec.gov/Archives/edgar/data/1156784/000135448809000768/corow10k.htm>>.

17. iRobot 10K Annual Report. Securities and Exchange Commission. 13 February 2009 13:47 UTC. 13 February 2009  
<<http://www.sec.gov/Archives/edgar/data/1159167/000095013509000964/b73459ice10vk.htm>>
18. "Autonomous robot." *Wikipedia, The Free Encyclopedia*. 3 Jul 2009, 05:45 UTC. 3 Jul 2009  
<[http://en.wikipedia.org/w/index.php?title=Autonomous\\_robot&oldid=300002856](http://en.wikipedia.org/w/index.php?title=Autonomous_robot&oldid=300002856)>.
19. "Robotics." *Wikipedia, The Free Encyclopedia*. 20 Aug 2009, 16:49 UTC. 20 Aug 2009  
<<http://en.wikipedia.org/w/index.php?title=Robotics&oldid=309093918>>.
20. "Robot." *Wikipedia, The Free Encyclopedia*. 24 Aug 2009, 03:23 UTC. 24 Aug 2009 <<http://en.wikipedia.org/w/index.php?title=Robot&oldid=309717516>>.
21. "A Robot in Every Home." Bill Gates. *Scientific American*. January 2007. 23 August 2009. <<http://www.scientificamerican.com/article.cfm?id=a-robot-in-every-home>>.
22. "Unmanned ground vehicle." *Wikipedia, The Free Encyclopedia*. 3 Aug 2009, 14:18 UTC. 3 Aug 2009  
<[http://en.wikipedia.org/w/index.php?title=Unmanned\\_ground\\_vehicle&oldid=305819838](http://en.wikipedia.org/w/index.php?title=Unmanned_ground_vehicle&oldid=305819838)>.
23. "Personal Robotics Industry set for massive growth." Mike Hanlon. *Gizmag*. 23 August 2009 01:55 UTC. 23 August 2009.  
<<http://www.gizmag.com/go/2407/>>.

24. Recording of MSNBC. Surveyor Corporation. 23 August 2009 01:55 UTC. 23 August 2009. <<http://www.surveyor.com/images/lunar-lander.mov>>.
25. "Skid steer." *Wikipedia, The Free Encyclopedia*. 19 Aug 2009, 19:50 UTC. 19 Aug 2009  
<[http://en.wikipedia.org/w/index.php?title=Skid\\_steer&oldid=308934370](http://en.wikipedia.org/w/index.php?title=Skid_steer&oldid=308934370)>.
26. "First-person shooter." *Wikipedia, The Free Encyclopedia*. 9 Aug 2009, 20:50 UTC. 9 Aug 2009 <[http://en.wikipedia.org/w/index.php?title=First-person\\_shooter&oldid=307037929](http://en.wikipedia.org/w/index.php?title=First-person_shooter&oldid=307037929)>.
27. "Personal Robot industry to grow to \$15 Billion by 2015." Noel McKeegan. *Gizmag*. 23 August 2009 01:55 UTC. 23 August 2009.  
<<http://www.gizmag.com/personal-robot-industry-to-grow-to-15-billion/8569/>>.
28. Appendix D
29. Appendix E
30. Appendix F
31. Appendix G
32. Press Release ECE/STAT/03/P01. United Nations Economic Commission For Europe. 21 October 2003 00:01 GMT. 21 October 2003.  
<[http://www.unece.org/press/pr2003/03stat\\_p01e.pdf](http://www.unece.org/press/pr2003/03stat_p01e.pdf)>.
33. [http://www.robotinvestments.com/RI\\_worldwide\\_market.htm](http://www.robotinvestments.com/RI_worldwide_market.htm)