Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139

Relationship Analysis Tool for Stakeholders in a Knowledge-Based Engineering Design System

An AUP Final Report submitted by:

Winston D Chang
6.199
Advanced Undergraduate Project Final Report
May 10, 2002

AUP Supervisor: Professor Amar Gupta

Abstract

This paper details the development and implementation of a relationship analysis tool for a knowledge-based approach to facilitating the engineering design process relating to spacecrafts. The relationship analysis tool focuses on the discovery and recognition of stakeholder relationships between one another and the design system. Most engineering design systems today involve the collaboration of multiple stakeholders over a large geographic distribution. The necessity of a relationship analysis tool is pertinent due to the essential requisite of understanding the dependencies and influences of all of the stakeholders of a system and the attribute scenarios of the design. An interactive interface allows for the implementation of functionality that allows for the analysis of stakeholder relationships and their respective attribute scenarios. This tool is built as an extension to a complimentary tool known as MIST, which facilitates the Multi-Attribute Tradespace Exploration process by enabling stakeholders to express their goals and preferences in a formalized manner.

Table of Contents

List of Figures

List of Equations

Acknowledgements

I would like to thank all of the people who have contributed to the development of the MIST (Multi-attribute Interview Software Tool) system and all those who have supported me throughout my work. This work would not have been possible without the support from the entire MIT SSPARC Information Technology development team. With all of their support and contributions, I have been able to develop, fully implement, and test my work. In particular, I wish to acknowledge Professor Amar Gupta and Satwiksai Seshasai for their constructive feedback and perceptive guidance throughout the development process.

1 Introduction

Today's engineering design processes typically involve multiple teams and multiple stakeholders collaborating together with the common objective to develop an engineering system most beneficial to each stakeholder's own initiatives. The complexity of designing an engineering system lies in the necessity of involving multiple stakeholders. Such an arduous task requires the utmost clarity in communication between all of the stakeholders involved, so that system requirements needed by each stakeholder are known by everyone. In addition to communication between stakeholders, all major design decisions should be documented and the rationale behind them recorded to ensure future understanding of decisions made in the past. To further the design process towards producing better higher-level decisions, it is especially vital to capture and analyze as much knowledge as possible regarding the systems under review. A main aspect to the understanding of complex engineering systems lies in the analysis of the relationships between each of the system stakeholders and the system itself. This Advanced Undergraduate Project details the implementation of a multi-stakeholder relationship analysis tool that allows for the analysis of stakeholder dependencies and system attributes.

In a traditional design environment, knowledge transfer was accomplished through informal contacts where design decisions and initiatives were addressed. Through emerging technologies, software tools have been constructed that assist in the design process to collaborate, share objectives, and record design rationales. By automating an interface for the design process, a knowledge repository of each specific design decision can be constructed. Such an interface not only assists in recording a large

repository of design rationale but also works to facilitate knowledge discovery by immediately processing and analyzing the data presented. This data mining of key knowledge regarding the design process has the potential to revolutionalize the engineering design process.

Currently, research being conducted as part of SSPARC (Space Systems, Policy, and Architecture Research Consortium) in the Department of Aeronautics and Astronautics at MIT addresses the topic of a knowledge-based approach to reducing impediments to collaboration amongst multiple stakeholders in a distributed design environment. The group has been specifically examining space system designs with the aim of producing optimal methods for selecting between various alternatives in space system architecture.

Recognizing the importance of stakeholder needs to the design process, an enterprise was undertaken by the PROFIT (Productivity from Information Technology) initiative at the Sloan School of Management at MIT, in conjunction with SSPARC, to develop a project that focuses on developing methods for collecting stakeholder desires and rationale and analyzing the utility and expense values for determined designs in an attempt to understand and capture system design rationale. This initiative, known as the Multi-Attribute Tradespace Exploration (MATE) process provides a formalized means of exploring a tradespace through the incorporation of preferences into design decision criteria with methods developed in economic and operations research theory. The MATE process specifically deals with the issue of capturing and analyzing the goals and utility of stakeholders in a design process. MIST (Multi-attribute Interview Software Tool) is the software tool that facilitates the MATE process by enabling stakeholders to express

their goals and preferences in a formalized structure. MIST accumulates stakeholder specific data from the design process and assembles this data with knowledge collected by the previously designed SSPARCy tool.

With the ability to gather and analyze system and stakeholder data from SSPARCy and MIST, the key to fully developing a knowledge-based engineering design tool lies in the ability to create knowledge from this data repository through the implementation of data mining technology. One of the major knowledge discovery efforts is the analysis of data from multiple stakeholders and design alternatives, to determine if patterns can be discovered relating how the design is affected by the goals and needs of stakeholders. The relationship analysis tool will determine the multi-attribute utility and expense values for all stakeholders and use this information as a valuation metric for relationships between stakeholders. The importance of this extension of MIST lies in the intricate dependencies between stakeholders in most engineering design projects. By understanding the effects on utility and expense values for stakeholders, the engineers of a system would then be best educated to develop optimal attribute scenarios for a system. This Advanced Undergraduate Project details the development of such a relationship analysis tool for a knowledge-based engineering design system as an extension to MIST.

2 Background

2.1 PROFIT Initiative

To address the notion of a knowledge-based approach for the analysis and capturing of system design rationale, the PROFIT initiative at the Sloan School of Management at MIT, headed by Professor Amar Gupta, has developed two systems that attempt to facilitate collaboration amongst multiple stakeholders in a distributed design environment. The first system, SSPARCy, acquires knowledge about a system design through the accumulation of knowledge regarding design rationales for main system decisions. The other system, MIST, establishes a framework to assist stakeholders in deciding the utility and expense functions regarding certain design alternatives through the eliciting and collecting of stakeholder goals and desires. Both of these systems capture the essential parts of the communication process that typically depends on geographic proximity and reduces these impediments to collaboration. The systems also incorporate the use of a four faceted knowledge-based approach of knowledge acquisition, discovery, management and repository to focus on various areas of implementation to be used in a specific design process [1].

With the basic functionality of SSPARCy and MIST implemented, the key to fully developing a knowledge-based engineering design tool lies in building intelligent functionality to process the captured knowledge. This Advanced Undergraduate Project details the analysis of data from multiple stakeholders and design alternatives, to determine trends and discover the relations between system design and the goals and needs of stakeholders. This relationship analysis tool is implemented to emphasize the importance of taking into account multiple stakeholder concerns with design decisions.

Such knowledge has become extremely valuable in a day and age where multiple stakeholders are often involved together, collaborating on a single design initiative.

2.1.1 SSPARCy Project

The SSPARCy project was designed to capture the rationale concerning major design decisions with the intent to allow for better recognition of future design trends and rationale [2]. The SSPARCy system provides efficient access to information regarding MATLAB source code files and the analysis of their interactions through graphical displays of the information collected. In addition to graphically displaying the knowledge collected, the tool is able to record the states of the code throughout the design process, perform integration integrity checks of code, and aid in the acquisition of design rationale for major system decisions. This myriad of functionality allows the system to support the process of design rationale capture.

2.1.2 MIST Project

The other tool developed by PROFIT facilitates the MATE process and is called MIST. This project is able to capture the goals and desires of stakeholders in a design process by using utility and expense analysis methods to obtain the relevant information regarding stakeholder interests in a system. Using the Multi-Attribute Utility Analysis (MAUA) interview process, which was developed by the SSPARC team, as a foundation, MIST employs an advanced Graphical User Interface (GUI) and a web-based computer interface to speed up and enrich the utility interview process for engineering systems. In MIST, the designer of the engineering system is able to describe system attributes, ranges for the values, units, and scenarios with input from stakeholders of the system.

The principle procedure for the capturing of stakeholder information is by conducting a series of interviews regarding stakeholder preferences of various attributes in a system design. From here, the collection of stakeholder utility and expense values for various design architectures can be analyzed to determine trends in a stakeholder's decision process and how a certain attribute set may relate to other stakeholders and their utility and expense values.

2.1.3 Four Faceted Knowledge Based Approach

Both tools described above achieve the goal of facilitating the design process through the use of the four faceted knowledge-based approach of knowledge acquisition, discovery, management and repository developed by the PROFIT initiative in conjunction with SSPARC [3]. This approach relies on the fundamental notion that knowledge is best used within a design process when it is incorporated to more effectively achieve engineering goals. The four facets are detailed below, and their respective applications to SSPARCy, MIST, and the relationship analysis tool are also described.

• **Knowledge Acquisition -** process of capturing information from various media, such as people's minds and handwritten documents, into computer accessible media

    • **SSPARCy** – GUI interface allows for simple data entry with visual indications of areas where more data is needed

    • **MIST** – interview process for utility and expense function generation is streamlined through implementation of process through software

• **Relationship Analysis Tool** – utility and expense data provided by MIST and properties and attributes provided by SSPARCy

• **Knowledge Management** – process of dealing with mitigating issues relating to differences in underlying contexts of information coming from dissimilar sources such as multiple stakeholders, multiple projects, and multiple stages of the process.

• **SSPARCy** – interface to process and review variable data with a history of all system changes

• **MIST** – history of all decision rationale concerning attribute changes and utility and expense preferences

• **Relationship Analysis Tool** – manages history of all attribute scenario changes and the resulting changes in utility and expense values for multiple stakeholders

• **Knowledge Discovery** – process of using emerging techniques to analyze large amounts of information to obtain better insights into information than otherwise would be possible using only human domain experts

• **SSPARCy** – identifies common characteristics between multiple design processes and uses links between design rationale and design decisions to predict future alternatives

• **MIST** – automatically generates utility and expense functions based on attributes of a design system for multiple stakeholders and design possibilities

• **Relationship Analysis Tool** – determines and shows the relationships between utility and expense values for each stakeholder and also the relationship of these values to an attribute scenario

• **Knowledge Dissemination** – process of automating the extraction of most relevant pieces of information from a large computer based information infrastructure with the extraction being tailored to the needs of different constituencies of users in an organization

> • **SSPARCy** – analyzed information can be used to assess design decisions and to help propose future initiatives
>
> • **MIST** – allows interaction of multiple stakeholders in delivering feedback regarding the factors that influence utility and expense for a system
>
> • **Relationship Analysis Tool** – generated relationships between stakeholder utility and expense values and generated attribute scenarios are analyzed to provide knowledge regarding stakeholder dependencies

This four-faceted technique allows for the efficient exchange of knowledge vital to collaboration [3].

3 Existing Work

The following projects are currently being pursued to facilitate the process of relating stakeholder goals and rationale to system attributes for the development of actual design parameters. A few of these projects also highlight functionalities similar to the attributes of the relationship analysis tool developed in this Advanced Undergraduate Project.

3.1 Rule-Based Algorithms Developed at Chung-Hua University

Researchers at the Chung-Hua University in Taiwan are developing systems that incorporate rule-based algorithms for relating customer needs and design specifications [4]. The researchers are transferring the customer needs directly into specifications by developing a matrix of weights between attributes and design factors. These weights are then used to determine relations that indicate how certain design parameters should change based on other parameter values.

The MIST tool extrapolates on this methodology by incorporating the goals and desires of multiple stakeholders in addition to attributes and design parameters. For the analysis of multi-stakeholder relations, the MIST relationship analysis functionality calculates multi-attribute utility and expense values to determine relations between stakeholders and attribute scenarios. The systems developed at Chung-Hua University does not incorporate multi-attribute utility and expense values into the analysis of attribute relations. Instead, their system focuses more on the interdependencies between engineering design parameters.

3.2 Questmap

Questmap is a commercially available tool whose functionality allows for it to capture design decision rationale. The Soft Bicycle Company creates the tool, and it focuses on recording the rationale behind design decisions [5]. The tool creates templates that can be used to structure design rationale through the use of icons and relationship arrows.

The design of Questmap allows for the creation of relationship maps, but unlike MIST, Questmap fails to discover new knowledge of the system through relationship analysis. Instead of determining the consequences of stakeholder relations, Questmap statically displays the relationships of design decisions with visual arrows that qualitatively show relationships. The relationship analysis tool of MIST generates the multi-attribute utility and expense values of stakeholders and relates them through calculated attribute scenarios, therefore creating knowledge regarding stakeholder dependencies.

3.3 DICE

The DICE (Distributed and Integrated Collaborative Engineering Environment) methodology was designed by Ram Sriram and offers a platform for collaborative engineering [6]. The system sections an engineering project into a set of modules and allows for work to be conducted simultaneously on each module. In the event that parallel work on the system results in conflicting decisions about a design aspect, a design rationale would be used to negotiate an optimal outcome between the engineers.

The MIST tool enhances the design rationale of a system by providing a history of design parameters and their associated utility and expense functions. The relationship

analysis tool would further improve the design rationale of a system by allowing for the determination of stakeholder relationships to attribute scenarios of the system. Currently the DICE methodology does not implement sophisticated stakeholder relationship functionality.

3.4 CHIC

The Center for Health Informatics and Computing (CHIC) has been incorporating socio-technical approaches for addressing issues with healthcare that necessitate the integration of information systems. The use of Soft Information Systems and Technology Methodology (SISTeM), participative simulation modeling, and stakeholder analysis have allowed for CHIC to recognize the necessity for an integrated socio-technical initiative [7]. This system uses a multiple stakeholder analyses to determine the importance of various stakeholders to the information system.

This system incorporates the most extensive relationship analysis tool out of all current research initiatives. However, unlike the relationship analysis tool of MIST, CHIC does not incorporate multi-attribute utility and expense values in the measure of relationships between the stakeholders and the system.

3.5 Utility Evaluation at the University of Massachusetts

Researchers at the University of Massachusetts in the Trade-off Based Robust Modeling and Design group have developed a tool that allows for online utility interviews to be conducted that determine stakeholder preferences [8]. The system also incorporates functionalities that allows for it to address issues of preference consistency and uncertainty during the interview process.

The interview process implemented by the MIST tool is very similar to the one pursued by the researchers at the University of Massachusetts. MIST takes the idea of an evaluation tool a step further with additional mechanisms such as the relationship analysis tool, something that the University of Massachusetts system is currently incapable of doing.

3.6 ICAD System

The ICAD system is widely used by world-class manufacturers as a knowledge based engineering tool that automates much of the system-level design and product design. The system is currently used by many large aerospace, automotive, and other intensive engineering manufacturers such as Boeing, Ford, etc [9]. By automating certain aspects of the engineering design process, ICAD is able to conduct additional design rationale and knowledge capture of the system.

Engineering systems employ a large group of engineers who are often located in various locations throughout the world. Such design systems would greatly benefit from a tool such as MIST that would be able to complement existing engineering tools such as ICAD. The relationship analysis tool for MIST would positively influence engineering systems with the ability to create knowledge regarding the relationships between multiple stakeholders of engineering systems.

4 Project Description

4.1 Overview of Motivation for Relationship Analysis

With the basic functionality developed to capture system design and stakeholder knowledge representations, the implementation of data mining technologies was necessitated. This Advanced Undergraduate Project extends the value of the MIST system by providing the functionality to compare the preferences of similar stakeholders in an engineering design process. This relationship analysis tool will enable designers to examine the relationships and dependencies of each stakeholder's utility and expense values to the attributes of a system's design. This would also allow for the discovery of knowledge concerning the optimal attribute set that would maximize utility and expense for each stakeholder. Figure 1 shows an example of an engineering design project that employs multiple stakeholders. From the diagram, it is shown how stakeholders of a system are all related to the project and will all influence the final outcome of the project.
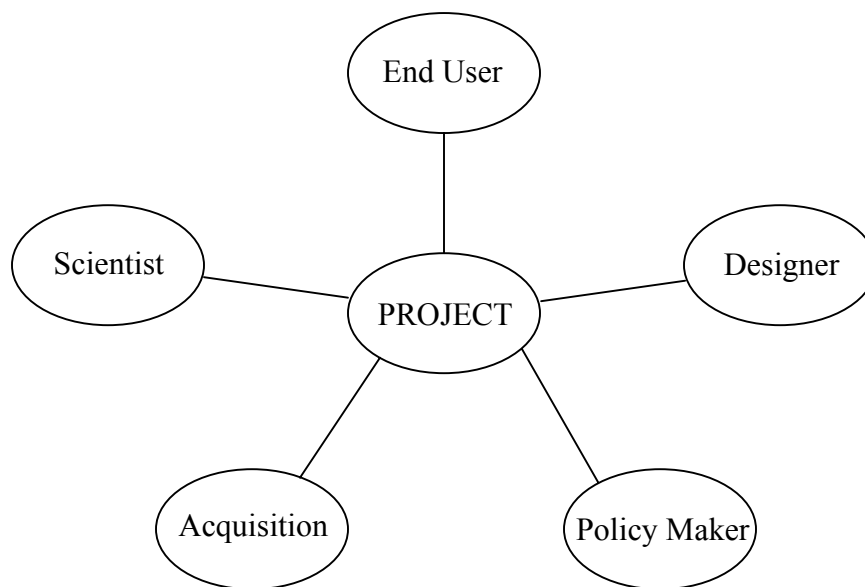

Figure 1: Multiple-Stakeholders for a Project

The motivation for the relationship analysis tool lies in the value added of the understanding of the relationships between stakeholders of a system. The relationship analysis tool creates knowledge discovery in two ways. First, the tool can analyze and calculate utility and expense values for each stakeholder of a system design based on a given set of design attributes. Second, by taking into account the utility and expense values of a stakeholder, the tool can generate and display the attribute set that generates the values for that particular stakeholder and would then show all correlating utility and expense values for every other stakeholder based on the attribute set that is found.

With the relationship analysis tool, the consequences of each design decision to system attributes can be recorded based on the changes to the utility and expense functions for all of the stakeholders in the system. The tool captures the evolution of the system design by recording all changes to the system attributes and relates each change to the resulting changes of stakeholder utility and expense. It is the capturing of this data and the effects of design decisions on attribute scenarios and utility and expense values that adds value to the MIST system. Relationships between various roles in the design process are better explained through their utility and expense data, and future projects benefit from knowing how past projects in the same tradespace operated under similar conditions.

This research is intended as a focal point for future initiatives into assisting the engineering design process. MIST is intended to be used by stakeholders to maximize the utility of a design process for everyone involved. The relationship analysis tool is intended to add to the functionality of dynamic representation of dependencies between stakeholders. Major changes in attribute scenarios will result in changes in multi-

attribute utility and expense values, and the resulting trends and analyses of these changes in values will allow system designers to understand tradeoffs in system design. With a strong understanding of stakeholder dependencies and relations, systems can be designed to optimize utility and expense values for all stakeholders involved.

4.2 Relationship of Properties to Attributes

Design system properties are the actual system parameters of the engineering design process. The properties are the values of all the system specifications, and are adjusted based on design requirements. Attributes are defined characteristics of a project that describe the important factors for stakeholders. Attributes are changed by modifications to system properties, and the two values are related through complex relations that are currently modeled as MATLAB functions. Property values change through the design process as the system designers change design specifications. Attribute values only change due to changes in property values that then effect the changes in the respective attributes.

The design variable for the current system being evaluated by MIST can be sectioned into four groups, Orbit Related, Spacecraft Related, Launch/Operations Related, and Program Related. Within each group, there are properties that are all related to the certain design aspect that the group is focused on.

4.3 Relationship of Attributes to Stakeholder Utility and Expense

The system utility and expense values for each stakeholder are determined through a mathematical analysis of the attributes of the system. All of the attributes of a specific design system belong to the same attribute set, or attribute scenario. Within an

attribute scenario, even the adjustment of one attribute value may generate new utility and expense value for some stakeholders.

Though an attribute scenario contains multiple attributes, each individual stakeholder is influenced by different sets of the attributes. For instance, a Scientific stakeholder may be influenced by Knowledge/Acc Altitude, Mission Lifetime, Time Spent in Region, Latitude Range, Latency, etc., while an Acquisition stakeholder is influenced by Cost, # of Additional Missions, Lead Time, Risk, etc. Since there are so many attributes of a system, each stakeholder is not necessarily influenced by every attribute because every stakeholder has differing goals and desires that are met by only a select group of a system's attributes. Figure 2 below shows an example of the relation of an Acquisition stakeholder to its respective attributes.
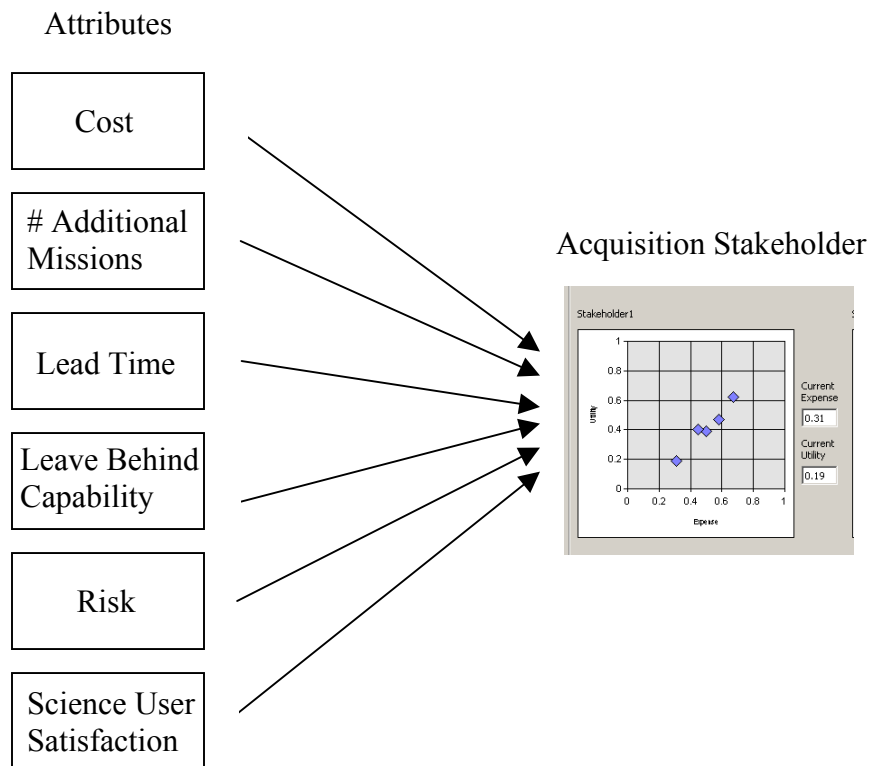
Figure 2: Relationship of Attributes to Values for Acquisition Stakeholder

The relationship analysis tool for MIST focuses on the analysis of the relationship between system attributes and stakeholder utility and expense values. The analysis tool generates utility and expense values for all stakeholders with a specific attribute sets and also generates the corresponding utility and expense values for stakeholders based on a calculated attribute scenario defined by a given utility and expense value of a specific stakeholder. By understanding the relationships between system attributes and stakeholder utility and expense values, the designers of a system will be able to optimize overall utility and expense values for the system.

5 Project Implementation

5.1 VBA Platform

The relationship analysis tool is employed through the GUI interface for MIST. The tool is implemented in Microsoft Excel and coded in Visual Basic for Applications (VBA). Excel was used as a base because of the ubiquitous nature of the application and the familiarity of the interface for most users. Excel provides pre-existing GUI functionality and familiar data analysis tools that can be employed by most stakeholders of the design system. The advantage of building off of a familiar GUI is that this allowed for more time to be devoted to the development of complete functionality of tools and initiatives instead of concentrating on the interface.

5.2 Relationship Tool Design Layout

In order to facilitate the functionality of relationship analysis, a modular approach to stakeholder relationships was undertaken. Below, the architecture of the relationship analysis extension of MIST is carefully outlined.

5.2.1 Stakeholder Analysis Main Form

The core aspect of the relationship analysis tool is the main form. This form was designed as the main interface and data display of the tool. The form displays the names and current values of all system properties and attributes in addition to all utility and expense values of each system stakeholder for every attribute scenario considered. It is important to list current properties and attributes so that the system architects can understand the relationship between the properties and attributes. All utility and expense values of each stakeholder for every considered attribute scenario are displayed to show

how design attribute changes have effected utility and expense values for each stakeholder. Figure 3 shows a screenshot of the main form of the relationship analysis tool.
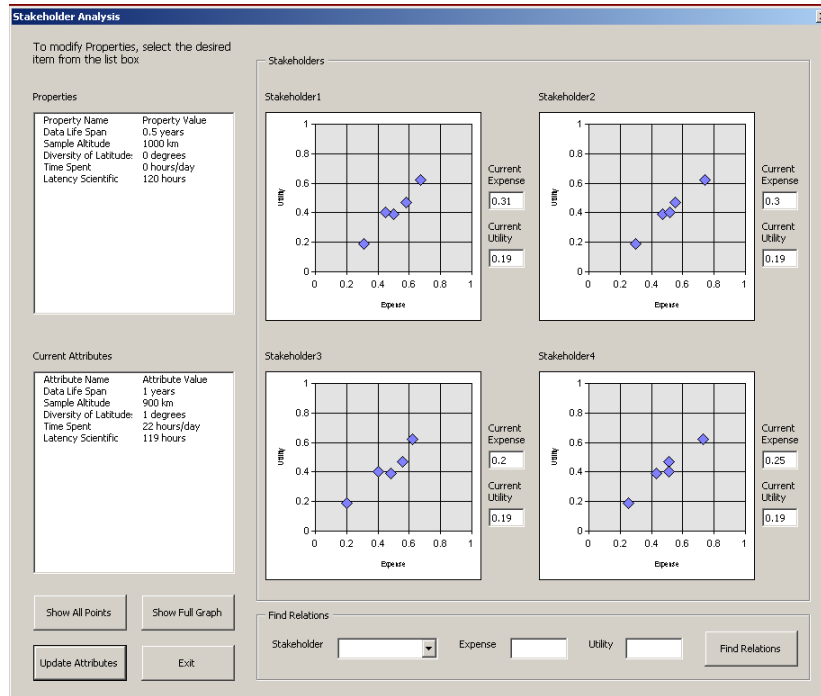
Figure 3: Main Form of Relationship Analysis Tool

The main form contains two sectioned framed areas. The top framed area is named "Stakeholders," and this frame contains all the graphical representations of the utility and expense curves for each stakeholder. The bottom framed area is named "Find Relations," and this area contains the functionality that drives the knowledge discovery of the relationships between stakeholder utility and expense values with attribute scenarios. The attributes and properties are listed in their respective list boxes on the left side of the form.

The utility and expense values for stakeholders are driven by the attribute values. In order to calculate the multi-attribute utility value and multi-attribute expense value for a stakeholder, a weighted average of the attributes is incorporated. This is further explained in section 5.2.5.

The functionalities of the relationship analysis tool are driven by the five buttons on the main form. With these buttons, the user can show all points, show the full graph, update attributes, exit the form, and find relationships of stakeholders.

5.2.2 Attribute Scenario Update Form

Updating of attribute scenarios is conducted through the implementation of the scenario update feature. Figure 4 shows a screenshot of the "Attribute Scenario" form that is accessed through the "Update Attributes" button on the main form. The "Attribute Scenario" form displays the current attribute scenario with the functionality of updating individual attributes and updating the entire attribute scenario when all the individual attribute changes are made. The attribute listing is generated from a table in the Excel workbook for the tool.
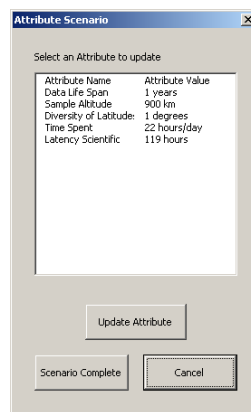
Figure 4: Attribute Scenario Form

The ability to update attributes as scenarios allows for the user to make multiple attribute changes at one design stage without having to recalculate utility and expense values for each respective design change. Attribute scenarios section the design process into various design states and allow the user to visualize the entire design development process.

5.2.3 Attribute Update Form

Attributes are finally updated with the "Attribute Update" form. Here the selected attribute and the current value for it are listed in the label on the form. Also listed are the maximum and minimum values for the selected attribute so that the user will not enter in values that are out of range. Figure 5 shows the layout of the "Attribute Update" form.
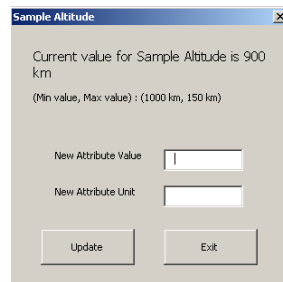
Figure 5: Attribute Update Form

The user is prompted to enter in a new attribute value in addition to the units for the value. The attribute values and units are entered in separately because they are stored separately in data columns in Excel. An Excel worksheet is where all of the history of attribute scenarios and stakeholder utility and expense values are kept. This is further described in section 5.2.6 of the paper.

5.2.4 Stakeholder Attribute Relations

The knowledge discovery aspect of the relationship analysis tool is mainly encompassed with the "Find Relations" section of the tool. Here, a stakeholder's specific utility and expense value can be entered and the attribute scenario that calculated those values will then be generated and displayed. Utility and expense values for all other stakeholders are then calculated based on the attribute scenario found. Figure 6 below shows an example result screen from the use of this relationship analysis functionality.

Figure 6: Find Relations Functionality

The utility and expense graphs for each stakeholder now list the specific values that are generated by the specific attribute scenario responsible for the utility and expense values of the stakeholder that was inputted in the "Find Relations" frame. This specific attribute scenario is also displayed in the attribute listbox. This feature is important

because it allows for users of the tool to understand the relationships between stakeholder utility and expense values for specific attribute scenario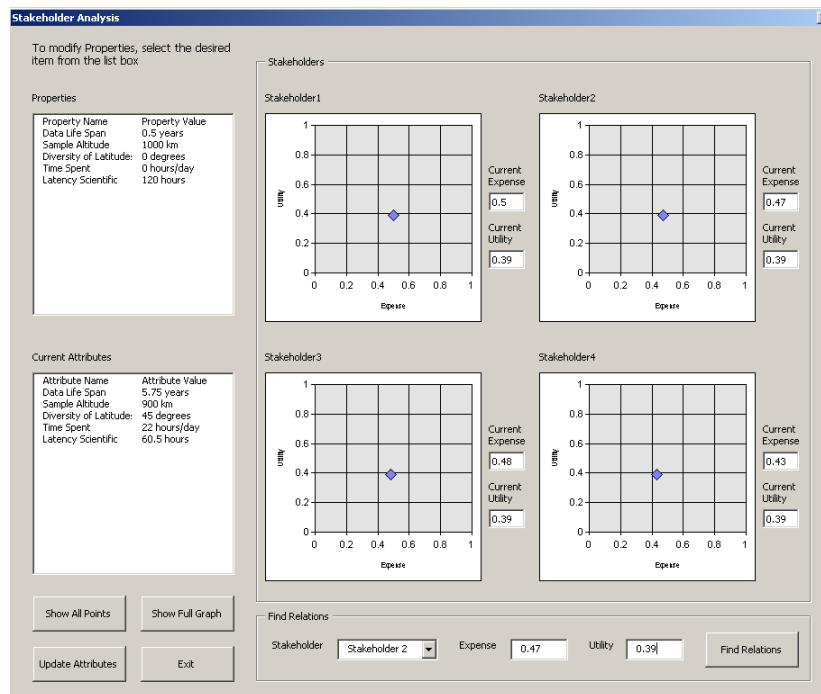s. With this tool, trends can be determined that show the progress and state of utility and expense at various attribute scenarios.

The way that this is done is that the tool takes the stakeholder selected and the utility and expense values entered and searches for matching utility and expense values for that stakeholder in the table in worksheet 'Test'. Once it finds the correct row where the values for the selected stakeholder is found, the index number of the attribute scenario is taken from the first cell in that row, and the attribute scenario is identified by the index number in the attribute scenario table. From here, the attributes of the index number selected are loaded onto the form, and the respective utility and expense values for that attribute are loaded into the chartspaces and the respective textboxes.

To return the state of the main form to show the current attribute scenario and all values for all the stakeholders, the "Show All Points" button can be used to return the main form to the current content available for it. This allows the user to return to the initial state from which all commands can be enacted again. The tool reloads from the latest attribute scenario and updates the charts with all utility and expense values.

In order to determine the values of utility and expense points that are used to determine relations of stakeholders, the specific values from the graph must be known. The chartspaces on the main form unfortunately are not able to the display values of the individual points on the graph, so another alternative was needed in order to realize the point values.

To accommodate this, the "Show Full Graph" functionality allows for a chart to be inserted into the Excel workbook containing a full sized chart of a specified stakeholder utility and expense graph. In order for the tool to recognize which stakeholder chart to pull up, the stakeholder must be specified in the combobox in the "Find Relations" frame. The chart that is then created is valuable to the user, because the chart displays the values of points when a mouse-over occurs on the point in question. With this user functionality, the user can determine the exact value of the points with which relations will be computed from. An example of the inserted chart is seen in Figure 7.
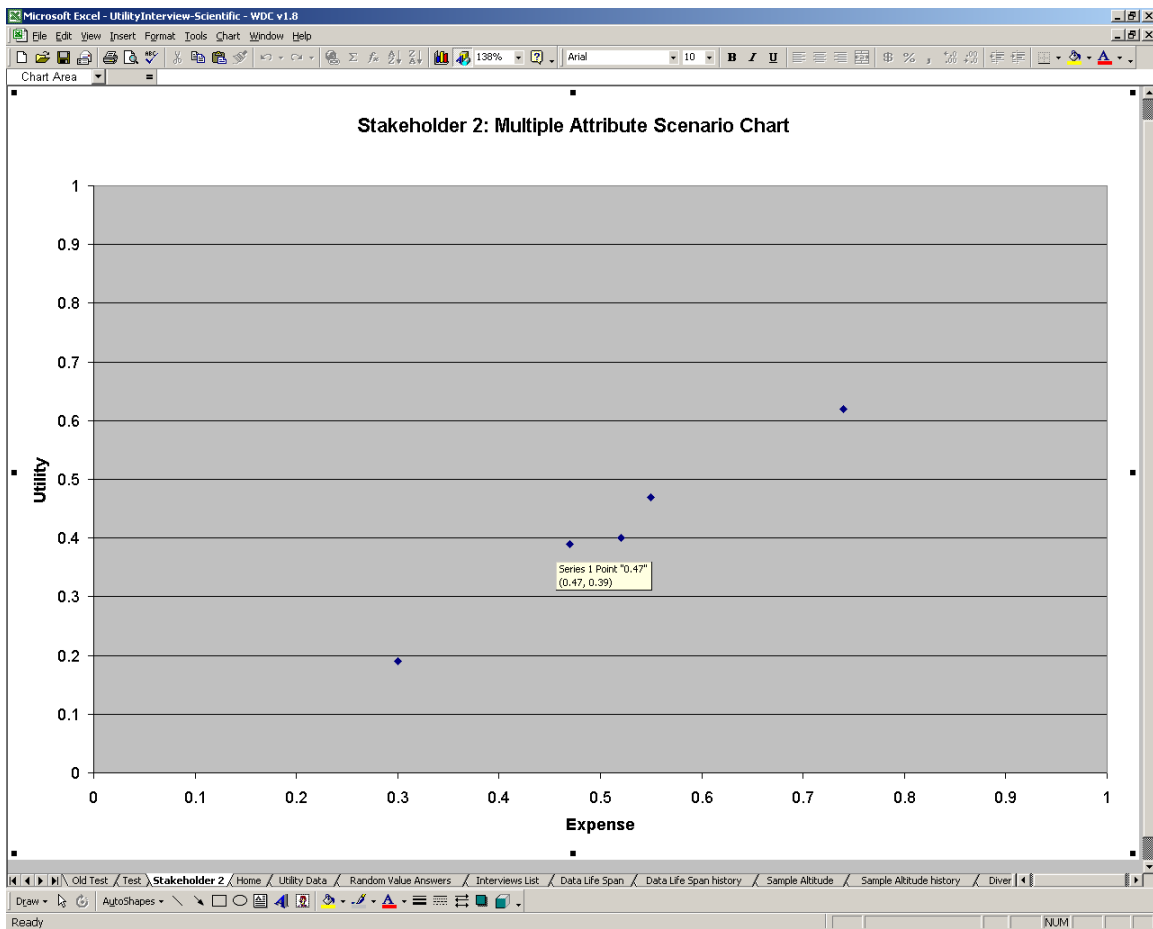
Figure 7: Stakeholder Chart Inserted in Workbook

Each time that the "Show Full Graph" button is used, the tool will automatically ask if the previous stakeholder chart inserted in the workbook, if there is one, needs to be deleted. This functionality is in place to ensure that the workbook does not become overpopulated with inserted stakeholder charts.

5.2.5 Multi-Attribute Utility and Expense Calculations

The utility and expense values for each stakeholder are influenced by multiple attributes. To calculate the multi-attribute utility values for each stakeholder, a time-weighted average of the attributes is used. Equation 1 shows the mathematical equation for this relationship.

$$KU(\underline{X})+1 = \prod_{i=1}^{N}\left(Kk_iU(X_i)+1\right)$$

Equation 1: Multi-Attribute Utility Equation [10]

In the equation, 'K' is the normalization constant and 'U($\underline{X}$)' is the multi-attribute utility function value. 'k_i' is the corner point value or relative weight of a specific attribute, and 'U(X_i)' is the value for a single attribute utility. The multi-attribute utility value is 'U($\underline{X}$)', and it is calculated by subtracting Equation 1 by one and then dividing by 'K', the normalization constant.

The utility value for the single attribute, 'U(X_i)', is extrapolated from the utility curve that is generated from the single attribute interviews. The single attribute interview builds a utility function for each attribute by extrapolating a graph from the calculated utility of N particular attribute values. The number of tested attribute values, N, is user-defined. This is shown in Figure 8, below, where the single attribute utility curve for the

attribute Diversity of Latitude is shown. In this example, seven particular attribute values are interviewed for, and the resulting curve is generated. It is from the lines connecting the known attribute values from which the single attribute value, 'U(X_i)' is determined.
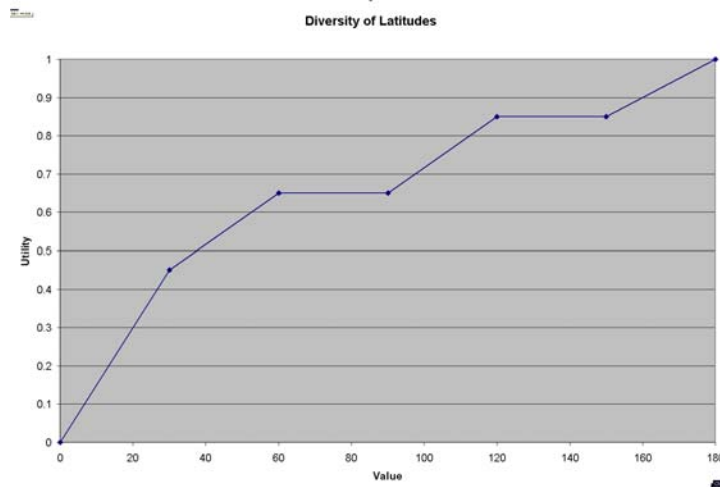
Figure 8: Single Attribute Utility Curve

To find the utility value for the single attribute, 'U(X_i)', the attribute value from the scenario is taken and matched up to the utility curve attribute values to find the point that is just greater than one of the calculated attribute values. This point and the point before it on the utility curve are taken, and a line is calculated to run through these two points. Now, the attribute value is plotted on this line to produce a utility value for the single attribute. The utility values are then stored in a one-dimensional array for future reference to determine multi-attribute utility values.

The corner point values are calculated from the multi-attribute corner point interviews. The values of these 'k_i' give a good indication of the relative importance between the attributes and act as a sort of weighting system. These corner point values

are found on each attributes' individual datasheets. The corner point values are then stored in a one-dimensional array for future reference.

The scalar 'K' is a normalization constant that ensures the multi-attribute utility function has a zero to one scale. 'K' can be calculated from Equation 2 below and must be $-1 < K < 1$ and $K \neq 0$.

$$K + 1 = \prod_{i=1}^{N} (Kk_i + 1)$$

Equation 2: Calculate 'K' Constant Equation

To calculate 'K', the 'k_i' values are known to be the corner point values, and the right side of Equation 2 can be multiplied out to a polynomial. Now, if 'K+1' from the left side is subtracted from both sides, the resulting equation would be equal to zero. The equation is shown by Equation 3 below.

$$0 = (Kk_1 + 1)(Kk_2 + 1) \bullet \ldots \bullet (Kk_{i-1} + 1)(Kk_i + 1) - (K + 1)$$

Equation 3: 'K' Polynomial Equation

In order to solve this polynomial that is set to zero, the relationship analysis tool fills in all 'k_i' values with the values of the array storing the corner point values. Now, to find 'K', the tool solves the polynomial equation by guessing and checking by entering a specific value for 'K' in order to see how close the solution is to 0. The values used for 'K' range from -1 to 1 and increase with 0.1 increments. Every time that a 'K' value is tried, the tool will store the value for 'K' if the solution to the polynomial is one of the two solutions closest to zero for all 'K' value tried. To determine the solutions closest to

zero, the absolute values of the solutions is determined, and the smallest absolute value solution is the closest value to zero. The reason that the two smallest 'K' values are saved is so that another decimal value for 'K' will be determined from these two values. Once the two values are found, the smaller 'K is incremented up by 0.01 storing only the new calculated 'K' value with two decimals that is closest to zero. This is approximately the value of 'K' to the hundredth decimal place.

Now that the values for 'K', 'k_i', and '$U(X_i)$' are known, the multi-attribute utility value, '$U(X)$', can be calculated. To do this, Equation 1 is subtracted by 1 from both sides and divided by 'K' from both sides as well. By doing this, the '$U(X)$' value is isolated on the right side of the equation and with all the values for the variables known, '$U(X)$' is easily calculated. This algorithm is used to calculate all multi-attribute utility values.

To calculate the multi-attribute expense values, the same values were taken for the attributes respective utility, and then a random deviation was calculated and added to the utility value to get the expense value. The random deviation was generated as a random number from 0 to 0.15, and this is added to the attribute's multiple-attribute utility value to get the multiple-attribute expense value.

The application of multi-attribute utility analysis results in a mathematical process that can be used to analyze a large trade space. With an increase in the number of attributes of the system the complexity increases rapidly. However, it is difficult to capture trends of a complex system with a small number of attributes.

5.2.6 Test Worksheet

To drive the relationship analysis tool, the values of all attribute scenarios and utility and expense values need to be stored and easily accessible. To do this, two tables that store the data are created in an Excel worksheet called 'Test' in the MIST workbook. One of the tables stores all the attribute scenarios and their values, and the other table stores all utility and expense values for stakeholders based on the attribute scenario that generated them. Figure 9 below shows the two tables in the 'Test' Worksheet.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Attributes	Expense1	Utility1	Expense2	Utility2	Expense3	Utility3	Expense4	Utility4		Attributes			2
2	1	0.58	0.47	0.55	0.47	0.56	0.47	0.51	0.47		1	Data Life Span	5.75	years
3	2	0.67	0.62	0.74	0.62	0.62	0.62	0.73	0.62		1	Sample Altitude	575	km
4											1	Diversity of Latitudes	90	degrees
5											1	Time Spent	12	hours/day
6											1	Latency Scientific	60.5	hours
7														
8											2	Data Life Span	5.75	years
9											2	Sample Altitude	175	km
10											2	Diversity of Latitudes	45	degrees
11											2	Time Spent	2	hours/day
12											2	Latency Scientific	60.5	hours

Figure 9: Tables in 'Test' Worksheet

The table that stores attribute scenarios gives an index number to the group of attributes so that the scenario can be recognized together. The table has four columns, index number, attribute name, attribute value, and attribute unit. Each attribute is listed

with its value for that scenario as well as the index of the scenario. The index is kept with an index incrementer that is a cell on the worksheet that shows the current index number. This index number is incremented every time that an attribute scenario is added.

The table that stores the actual utility and expense value references the values associated with an attribute scenario using the index number related to the scenario. The table has nine columns. The first column is the attribute scenario's index number, and the next eight columns are the utility and expense values for the four stakeholders of the system. Every time that a new attribute scenario is added and the index is incremented, that index number is placed in the first column, and the respective utility and expense values for that attribute scenario fill in the rest of the row.

The two tables in the Excel worksheet were used to store the information needed for the relationship analysis tool because this is the most convenient way to display and access the needed information. By having a table solely for attribute scenarios and another table with related utility and expense values for all stakeholders, the tool can easily identify which utility and expense values correlate to which attribute scenarios.

5.3 Issues Encountered

5.3.1 User Form Layout

During the design of the relationship analysis tool, one of the main issues and design decisions involved the layout of the user forms. By using a main form from which all the functionality of the tool can be initiated from, the tool is able to have a central starting point. One of the main concerns for the tool was to be able to have a user interface that is easy to navigate and that displays all the relevant information to understand the relationships between stakeholders and the system.

The main form always displays the current design properties as well as the current attributes of the system. With these two sets of values displayed on the main form, users are able to understand where the current state of the system.

The functionality of separating attribute updating and the updating of attribute scenarios into two forms allows for the user to easily understand the role of attribute scenarios as design stages. Attribute scenarios are also recorded as a set of attributes in the Test worksheet, and this allows for the identifying of each particular attribute value.

5.3.2 Determining Utility and Expense Values From Stakeholder Graphs

One of the main requirements needed for the relationship analysis tool to work is the ability to recognize utility and expense values of a certain stakeholder that is of interest. By identifying the utility and expense value of a stakeholder, the attribute scenario that defined that specific value can be determined and displayed. Using this attribute scenario, the utility and expense values for the other stakeholders at this scenario can then be found as well. As shown, the entire process is driven by the ability to recognize specific points and their values, and this was a critical issue with which multiple alternatives were considered in hopes of resolving the concern.

One of the first considered solution to this problem was the implementation of interactive chart spaces on the main form. The idea was to make the graphs on the form interactive so that once the user recognizes a utility and expense point of interest, it could select the point by clicking directly on the point on the graph. Unfortunately, chart spaces in forms cannot be made interactive and do not respond to any mouse click commands.

Another alternative considered was the use of the mouse over command on the chart spaces on the form. The idea was that if the user was able to mouse over the intended point, the value of that point would be displayed, and the user could separately enter in this value. Unfortunately, the mouse over functionality does not exist for chart spaces on forms. This is related to the fact that chart spaces are inactive areas on a form.

Building off of the mouse over idea, the final implementation of this issue involved the functionality of creating a chart in the workbook of the tool. By creating a chart containing the graph of the stakeholder in question, the user can then mouse over the graph when it is a chart in the Excel workbook. This functionality allows for the user to recognize and positively identify what the values for the point in question are.

5.3.3 Calculating Multi-Attribute Utility and Expense Values

Another issue encountered was in the calculation of the multi-attribute utility and expense values. In the calculating of the utility value, two key mathematical assumptions were made. First, in calculating the single attribute utility values for the values of the attributes in the scenario, the utility value was taken from the utility curve generated by the single attribute interviews. Because the utility curve consists of only a limited amount of calculated points, chances are that the attribute value from the scenario is not one of the calculated points in the extrapolated utility curve. The assumption is made is that the utility curve is generated by modeling straight lines connecting each adjacent known utility value. In order to calculate the value, a line between the two known values around the value in question is made, and the value in question can be extrapolated from this line. This assumption of creating straight lines between points is the best resolution

to the issue of the extrapolated utility curve without having calculated every point of the utility curve.

Another assumption made while determining multi-attribute utility is during the calculation of 'K' the normalization constant. While calculating 'K', a polynomial results and needs to be solved. Without an adequate polynomial solver, a guess and check method was implemented to solve for the 'K' value to the hundredth decimal value. The algorithm to implement this is described in section 5.2.5. An alternative to this guess and check methodology would have been to use MATLAB and its polynomial solver to calculate the solution. However, the solution generated from the guess and check alternative proved to be just as good of an alternative because the solution is just as accurate to the hundredth decimal place as a MATLAB polynomial solver would have been.

Finally, the issue of calculating multi-attribute expense values proved to be an important concern for the relationship analysis tool. So far, the MIST system has not implemented the ability to calculate expense values. In theory, the algorithm to calculating expense values would be very similar to the algorithm used to calculate utility. The single interviewing and corner point interviewing processes would still be applied, but this time, they would be applied to determine expense. Even the calculation process of the multi-attribute value would incorporate the same weighted mathematical relationship. Due to the fact that the expense analysis has not been implemented, it was assumed that the best way to currently model expense for the relationship analysis tool would be as a deviation from the utility value.

5.4 Lessons Learned

Throughout the development process of the relationship analysis tool, it became apparent that the main issue for the tool to resolve would regard the determination of which data is most relevant to the user in determining the relationships between multiple stakeholders in a developing design environment. To accommodate the analysis of multiple stakeholders, it is important that the software interface is highly structured and easy to navigate. Much of the knowledge acquired by the system relies on the ability to specify which utility and expense values for a stakeholder need to be analyzed. Once selected utility and expense points are specified, the user is able to recognize the trends and values generated by certain attribute scenarios.

6 Extending Relationship Analysis Tool

6.1 Overview

The relationship analysis tool described in this paper reflects the initial design of the core functionality to foster knowledge discovery regarding relationships and stakeholder analyses of an engineering design system. The tool allows for the analysis and recording of the changes in utility and expense values of numerous stakeholders of a system due to changes in attribute scenarios. The tool also leads to knowledge discovery of relations between utility and expense values of various stakeholders for given attribute values.

From the core functionality, there are two extensions of the relationship analysis tool that are currently under consideration for development. The first extension involves the implementation of the functionality to accommodate the profiling of stakeholders. This will allow for users to review and analyze stakeholder utility and expense trends regarding engineering decisions. The second extension regards the implementation of a more dynamic user interface to allow for easier selecting and retrieval of stakeholder and attribute data.

6.1.1 Profiling of Stakeholder Analyses

The ability to recognize decision trends of stakeholders regarding utility and expense would be an important value added to the MIST system. Recording all utility and expense changes of a stakeholder during the design process and comparing the results to other projects, a stakeholder profile can be constructed. With the profile constructed, system engineers will be able to design systems that better accommodate the stakeholder's specific desires and goals.

6.1.2 Dynamic User Interface

The interface could be further enhances with the implementation of a more dynamic and informative layout. Future extensions would include additional forms illustrating additional analyses concerning stakeholder profiles and attribute scenario histories. It is important for the interface to be able to retrieve and display detailed information regarding the history of attribute scenarios and stakeholder utility and expense changes.

6.2 Comparison To Current Multi-Stakeholder Initiatives

Though there has been a lot of research regarding the implementation of a knowledge-based tool to facilitate collaboration in the engineering design process, however, there has been less emphasis on the relationship analysis of multiple stakeholders. Most research initiatives have been focusing on the analysis of a single stakeholder regarding an engineering system. The relationship analysis tool for MIST is the first tool that analyzes stakeholder relationships to each other and to the engineering system attributes in order to create knowledge to facilitate the design process.

By focusing on the multi-attribute and utility values for multiple stakeholders, MIST is really able to analyze stakeholder relations in order to build a better understanding of the dependencies between attribute scenarios and stakeholders. Systems such as the rule-based algorithms developed at Chung-Hua University did not incorporate the multi-attribute utility and expense values used by MIST into the analysis of attribute relations. This system instead focused on the interdependencies between engineering design issues. By focusing on the multi-attribute utility and expense values, MIST is able to create knowledge regarding multiple stakeholder relations.

Another instance of how the MIST application of multi-attribute utility and expense analysis is more sophisticated than existing systems can be seen with the comparison to CHICH. As mentioned earlier, the Center for Health Informatics and Computing (CHIC) has been incorporating multiple stakeholder analyses to determine the importance of various stakeholders to the information system of healthcare. MIST again better implements the analysis of stakeholders by incorporating multi-attribute utility and expense values in the measure of relationships between the stakeholders and the system.

The relationship analysis tool for MIST benefits most engineering systems that involve multiple stakeholders that may be geographically separated and relies on close interaction of dependent modules. The ICAD system used currently by many engineering designers is improved with the functionalities of MIST. ICAD like so many other systems fails to take into account the desires and goals of multiple stakeholders, and by combining the system with MIST, engineers would then be able to analyze and incorporate multiple stakeholder goals and needs to generate a better understanding of all design rationale.

7 Conclusion

With the development of the MATE process, MIST nicely fulfills the knowledge-based engineering design space as a tool that incorporates the ability to determine utility and expense values for stakeholders of a engineering design system. The relationship analysis tool outlined in this Advanced Undergraduate Project outlines the ability for a tool to facilitate knowledge discovery of gathered information. In this case, the tool is able to generate relationships and graphically depict the utility and expense values of all stakeholders at specific attribute scenarios. This functionality allows for the user of the tool to have a full understanding of the interactions of stakeholders and the system attributes in order to represent the thinking behind design rationale of a complex engineering system.

8 References

[1] Seshasai, Satwiksai. "Reducing Impediments to Collaboration in a Virtual Design World." MIT, Spring 2002.

[2] Scott, Quincy R. "SSPARCy: A Software Integration Support and Design Rationale Capture System." Master's thesis, MIT, July 11, 2001.

[3] Gupta, Amar. "A Four-Faceted Knowledge-Based Approach for Surmounting Borders." Journal of Knowledge Management: Vol. 5, No. 4, December 2001.

[4] Chiu-Chi Wei, Ping-Hung Liu and Chie-Bein Chen. "An Automated System for Product Specification and Design." Assembly Automation: Vol 20: pp 225-232. MCB University Press.

[5] QuestMap v3.12. The Soft Bicycle Company, 2000.

[6] Sriram, Ram D. <u>Distributed and Integrated Collaborative Engineering Environment</u>. Book to be published, 2002.

[7] Chris Atkinson, Tillal Eldabi, Ray Paul and Athanasia Pouloudi. "Investigating Integrated Socio-technical Approaches To Heath Informatics". Proceedings of 34th Annual Hawaii International Conference on System Sciences, 2001.

[8] Wan Jie and Krishnamurty, S. "Comparison based Decision Making in Engineering Design." ASME Design Theory and Methodology, Las Vegas, Nevada, 1999.

[9] Katragadda, Prasanna. "Knowledge-Based Systems Interoperability." Workshop 9, Concentra Inc.

[10] Diller, Nathan, et. al. "B-TOS: Terrestrial Observer Swarm." Final Report,

16.89 Space Systems Engineering, MIT, Spring 2001.

9 Appendix A: Code

9.1 Stakeholder Analysis Form Load Code

```
'Main function that loads the Stakeholder Analysis Form
Sub Load_StakeholderAnalysis()

  Dim att_list As Range
  Dim numrows As Integer
  Dim Att_range As Range
  Set att_list = Worksheets("Home").Range("Attribute_list")
  Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
  numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

  Dim Array_att(20) As String
  Dim i As Integer
  i = 0

  Do While i < numrows
     Array_att(i) = Att_range(i + 1).Value
     i = i + 1
  Loop

  Dim Array_A() As String
  ReDim Array_A(0 To numrows + 1, 0 To 2)
  Dim Array_B() As String
  ReDim Array_B(0 To numrows + 1, 0 To 2)
  Dim Array_C(20) As String
  Dim Array_D(20) As String

  Array_A(0, 0) = "Property Name"
  Array_A(0, 1) = "Property Value"
  Array_B(0, 0) = "Attribute Name"
  Array_B(0, 1) = "Attribute Value"

  If Worksheets("Test").Cells(1, 14) <> 0 Then
     Dim b As Integer

     b = 0
     i = 2

     Do Until (Worksheets("Test").Cells(i, 11) = Worksheets("Test").Cells(1, 14))
        i = i + 1
     Loop

     Do While b < numrows
        Array_A(b + 1, 0) = Worksheets("Test").Cells(i, 12).Value
        Array_C(b + 1) = Worksheets("Test").Cells(b + 1, 17).Value
        Array_D(b + 1) = Worksheets("Test").Cells(i, 14).Value
        Array_A(b + 1, 1) = Array_C(b + 1) & " " & Array_D(b + 1)
        Array_B(b + 1, 0) = Worksheets("Test").Cells(i, 12).Value
        Array_C(b + 1) = Worksheets("Test").Cells(i, 13).Value
        Array_B(b + 1, 1) = Array_C(b + 1) & " " & Array_D(b + 1)
        b = b + 1
```

```
        i = i + 1
    Loop

    StakeholderAnalysis.PropertiesListBox.List = Array_A
    StakeholderAnalysis.AttributesListBox.List = Array_B
End If


If Worksheets("Test").Cells(1, 14) = 0 Then

    Dim a As Integer
    a = 0

    Do While a < numrows
        Array_A(a + 1, 0) = Worksheets("Test").Cells(a + 1, 16).Value
        Array_C(a + 1) = Worksheets("Test").Cells(a + 1, 17).Value
        Array_D(a + 1) = Worksheets("Test").Cells(a + 1, 19).Value
        Array_A(a + 1, 1) = Array_C(a + 1) & " " & Array_D(a + 1)
        Array_B(a + 1, 0) = Worksheets("Test").Cells(a + 1, 16).Value
        Array_C(a + 1) = Worksheets("Test").Cells(a + 1, 18).Value
        Array_B(a + 1, 1) = Array_C(a + 1) & " " & Array_D(a + 1)
        a = a + 1
    Loop

    StakeholderAnalysis.PropertiesListBox.List = Array_A
    StakeholderAnalysis.AttributesListBox.List = Array_B

    'begin filling in of "Test" wkst
    Worksheets("Test").Cells(1, 14) = 1
    Worksheets("Test").Cells(2, 1) = Worksheets("Test").Cells(1, 14)

    a = 0
    i = 0

    Do While a < numrows
        Worksheets("Test").Cells(i + 2, 11) = Worksheets("Test").Cells(1, 14)
        Worksheets("Test").Cells(i + 2, 12) = Array_B(a + 1, 0)
        Worksheets("Test").Cells(i + 2, 13) = Array_C(a + 1)
        Worksheets("Test").Cells(i + 2, 14) = Array_D(a + 1)
        a = a + 1
        i = i + 1
    Loop

    Worksheets("Test").Cells(2, 2) = StakeholderAnalysis.CalcExpense("Stakeholder1")
    Worksheets("Test").Cells(2, 3) = StakeholderAnalysis.CalcUtility("Stakeholder1")
    Worksheets("Test").Cells(2, 4) = StakeholderAnalysis.CalcExpense("Stakeholder2")
    Worksheets("Test").Cells(2, 5) = StakeholderAnalysis.CalcUtility("Stakeholder2")
    Worksheets("Test").Cells(2, 6) = StakeholderAnalysis.CalcExpense("Stakeholder3")
    Worksheets("Test").Cells(2, 7) = StakeholderAnalysis.CalcUtility("Stakeholder3")
    Worksheets("Test").Cells(2, 8) = StakeholderAnalysis.CalcExpense("Stakeholder4")
    Worksheets("Test").Cells(2, 9) = StakeholderAnalysis.CalcUtility("Stakeholder4")
End If

Dim tmp As Integer

tmp = Worksheets("Test").Cells(1, 14) + 1
```

```
Worksheets("Test").Cells(tmp, 1) = Worksheets("Test").Cells(1, 14)

StakeholderAnalysis.UtilityTextBox1.Text = Worksheets("Test").Cells(tmp, 3)
StakeholderAnalysis.UtilityTextBox2.Text = Worksheets("Test").Cells(tmp, 5)
StakeholderAnalysis.UtilityTextBox3.Text = Worksheets("Test").Cells(tmp, 7)
StakeholderAnalysis.UtilityTextBox4.Text = Worksheets("Test").Cells(tmp, 9)

StakeholderAnalysis.ExpenseTextBox1.Text = Worksheets("Test").Cells(tmp, 2)
StakeholderAnalysis.ExpenseTextBox2.Text = Worksheets("Test").Cells(tmp, 4)
StakeholderAnalysis.ExpenseTextBox3.Text = Worksheets("Test").Cells(tmp, 6)
StakeholderAnalysis.ExpenseTextBox4.Text = Worksheets("Test").Cells(tmp, 8)

'graphing
    Dim Chart1 As owc.WCChart
    Dim Chart2 As owc.WCChart
    Dim Chart3 As owc.WCChart
    Dim Chart4 As owc.WCChart
    Dim Series1 As owc.WCSeries
    Dim Series2 As owc.WCSeries
    Dim Series3 As owc.WCSeries
    Dim Series4 As owc.WCSeries
    Dim XData1(1 To 40)
    Dim YData1(1 To 40)
    Dim XData2(1 To 40)
    Dim YData2(1 To 40)
    Dim XData3(1 To 40)
    Dim YData3(1 To 40)
    Dim XData4(1 To 40)
    Dim YData4(1 To 40)

    StakeholderAnalysis.ChartSpace1.Clear
    StakeholderAnalysis.ChartSpace1.Refresh
    StakeholderAnalysis.ChartSpace2.Clear
    StakeholderAnalysis.ChartSpace2.Refresh
    StakeholderAnalysis.ChartSpace3.Clear
    StakeholderAnalysis.ChartSpace3.Refresh
    StakeholderAnalysis.ChartSpace4.Clear
    StakeholderAnalysis.ChartSpace4.Refresh

    Set Chart1 = StakeholderAnalysis.ChartSpace1.Charts.Add
    Set Chart2 = StakeholderAnalysis.ChartSpace2.Charts.Add
    Set Chart3 = StakeholderAnalysis.ChartSpace3.Charts.Add
    Set Chart4 = StakeholderAnalysis.ChartSpace4.Charts.Add

    For i = 1 To 40
        XData1(i) = Worksheets("Test").Cells(i + 1, 2)
        YData1(i) = Worksheets("Test").Cells(i + 1, 3)
        XData2(i) = Worksheets("Test").Cells(i + 1, 4)
        YData2(i) = Worksheets("Test").Cells(i + 1, 5)
        XData3(i) = Worksheets("Test").Cells(i + 1, 6)
        YData3(i) = Worksheets("Test").Cells(i + 1, 7)
        XData4(i) = Worksheets("Test").Cells(i + 1, 8)
        YData4(i) = Worksheets("Test").Cells(i + 1, 9)
    Next
```

```
Chart1.Type = owc.chChartTypeScatterMarkers
Chart2.Type = owc.chChartTypeScatterMarkers
Chart3.Type = owc.chChartTypeScatterMarkers
Chart4.Type = owc.chChartTypeScatterMarkers
Set Series1 = Chart1.SeriesCollection.Add
Set Series2 = Chart2.SeriesCollection.Add
Set Series3 = Chart3.SeriesCollection.Add
Set Series4 = Chart4.SeriesCollection.Add

With Series1
    .SetData owc.chDimXValues, owc.chDataLiteral, XData1
    .SetData owc.chDimYValues, owc.chDataLiteral, YData1
End With

With Series2
    .SetData owc.chDimXValues, owc.chDataLiteral, XData2
    .SetData owc.chDimYValues, owc.chDataLiteral, YData2
End With

With Series3
    .SetData owc.chDimXValues, owc.chDataLiteral, XData3
    .SetData owc.chDimYValues, owc.chDataLiteral, YData3
End With

With Series4
    .SetData owc.chDimXValues, owc.chDataLiteral, XData4
    .SetData owc.chDimYValues, owc.chDataLiteral, YData4
End With

With Chart1.Axes(owc.chAxisPositionLeft)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Utility"
    .Title.Font.Size = 6
End With

With Chart1.Axes(owc.chAxisPositionBottom)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Expense"
    .Title.Font.Size = 6
End With

With Chart2.Axes(owc.chAxisPositionLeft)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Utility"
    .Title.Font.Size = 6
End With
```

```vba
    With Chart2.Axes(owc.chAxisPositionBottom)
        .Scaling.Maximum = 1
        .Scaling.Minimum = 0
        .MajorUnit = 0.2
        .HasTitle = True
        .Title.Caption = "Expense"
        .Title.Font.Size = 6
    End With

    With Chart3.Axes(owc.chAxisPositionLeft)
        .Scaling.Maximum = 1
        .Scaling.Minimum = 0
        .MajorUnit = 0.2
        .HasTitle = True
        .Title.Caption = "Utility"
        .Title.Font.Size = 6
    End With

    With Chart3.Axes(owc.chAxisPositionBottom)
        .Scaling.Maximum = 1
        .Scaling.Minimum = 0
        .MajorUnit = 0.2
        .HasTitle = True
        .Title.Caption = "Expense"
        .Title.Font.Size = 6
    End With

    With Chart4.Axes(owc.chAxisPositionLeft)
        .Scaling.Maximum = 1
        .Scaling.Minimum = 0
        .MajorUnit = 0.2
        .HasTitle = True
        .Title.Caption = "Utility"
        .Title.Font.Size = 6
    End With

    With Chart4.Axes(owc.chAxisPositionBottom)
        .Scaling.Maximum = 1
        .Scaling.Minimum = 0
        .MajorUnit = 0.2
        .HasTitle = True
        .Title.Caption = "Expense"
        .Title.Font.Size = 6
    End With

    FindRelations_Load

End Sub

Sub FindRelations_Load()
    StakeholderAnalysis.FRStakeholderComboBox.Clear
    StakeholderAnalysis.FRStakeholderComboBox.AddItem "Stakeholder 1"
    StakeholderAnalysis.FRStakeholderComboBox.AddItem "Stakeholder 2"
    StakeholderAnalysis.FRStakeholderComboBox.AddItem "Stakeholder 3"
    StakeholderAnalysis.FRStakeholderComboBox.AddItem "Stakeholder 4"
End Sub
```

9.2 Stakeholder Analysis Main Form Code

```
'Code that handles all event actions on the Stakeholder Analysis Form
Private Sub CommandButtonExit_Click()
    Unload StakeholderAnalysis
    StakeholderAnalysis.Hide
End Sub

Private Sub CommandButtonFindRelations_Click()
    Dim a As Integer

    Select Case FRStakeholderComboBox.Value
        Case "Stakeholder 1"
            a = 2
        Case "Stakeholder 2"
            a = 4
        Case "Stakeholder 3"
            a = 6
        Case "Stakeholder 4"
            a = 8
        Case Else
            MsgBox ("Error.  Please select a stakeholder.")
    End Select

    Dim i As Integer

    i = 2

    Do Until (Worksheets("Test").Cells(i, a + 1) = FRUtilityTextBox.Text) And
(Worksheets("Test").Cells(i, a) = FRExpenseTextBox.Text)
        i = i + 1
    Loop
    'now i contains the row number which the attribute set is at

    'GRAPHING
    Dim Chart1 As owc.WCChart
    Dim Chart2 As owc.WCChart
    Dim Chart3 As owc.WCChart
    Dim Chart4 As owc.WCChart
    Dim Series1 As owc.WCSeries
    Dim Series2 As owc.WCSeries
    Dim Series3 As owc.WCSeries
    Dim Series4 As owc.WCSeries

    StakeholderAnalysis.ChartSpace1.Clear
    StakeholderAnalysis.ChartSpace1.Refresh
    StakeholderAnalysis.ChartSpace2.Clear
    StakeholderAnalysis.ChartSpace2.Refresh
    StakeholderAnalysis.ChartSpace3.Clear
    StakeholderAnalysis.ChartSpace3.Refresh
    StakeholderAnalysis.ChartSpace4.Clear
    StakeholderAnalysis.ChartSpace4.Refresh

    Set Chart1 = StakeholderAnalysis.ChartSpace1.Charts.Add
    Set Chart2 = StakeholderAnalysis.ChartSpace2.Charts.Add
```

```
Set Chart3 = StakeholderAnalysis.ChartSpace3.Charts.Add
Set Chart4 = StakeholderAnalysis.ChartSpace4.Charts.Add

Chart1.Type = owc.chChartTypeScatterMarkers
Chart2.Type = owc.chChartTypeScatterMarkers
Chart3.Type = owc.chChartTypeScatterMarkers
Chart4.Type = owc.chChartTypeScatterMarkers
Set Series1 = Chart1.SeriesCollection.Add
Set Series2 = Chart2.SeriesCollection.Add
Set Series3 = Chart3.SeriesCollection.Add
Set Series4 = Chart4.SeriesCollection.Add

With Series1
   .SetData owc.chDimXValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 2)
   .SetData owc.chDimYValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 3)
End With

With Series2
   .SetData owc.chDimXValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 4)
   .SetData owc.chDimYValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 5)
End With

With Series3
   .SetData owc.chDimXValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 6)
   .SetData owc.chDimYValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 7)
End With

With Series4
   .SetData owc.chDimXValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 8)
   .SetData owc.chDimYValues, owc.chDataLiteral, Worksheets("Test").Cells(i, 9)
End With

With Chart1.Axes(owc.chAxisPositionLeft)
   .Scaling.Maximum = 1
   .Scaling.Minimum = 0
   .MajorUnit = 0.2
   .HasTitle = True
   .Title.Caption = "Utility"
   .Title.Font.Size = 6
End With

With Chart1.Axes(owc.chAxisPositionBottom)
   .Scaling.Maximum = 1
   .Scaling.Minimum = 0
   .MajorUnit = 0.2
   .HasTitle = True
   .Title.Caption = "Expense"
   .Title.Font.Size = 6
End With

With Chart2.Axes(owc.chAxisPositionLeft)
   .Scaling.Maximum = 1
   .Scaling.Minimum = 0
   .MajorUnit = 0.2
   .HasTitle = True
   .Title.Caption = "Utility"
```

```vba
    .Title.Font.Size = 6
End With

With Chart2.Axes(owc.chAxisPositionBottom)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Expense"
    .Title.Font.Size = 6
End With

With Chart3.Axes(owc.chAxisPositionLeft)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Utility"
    .Title.Font.Size = 6
End With

With Chart3.Axes(owc.chAxisPositionBottom)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Expense"
    .Title.Font.Size = 6
End With

With Chart4.Axes(owc.chAxisPositionLeft)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Utility"
    .Title.Font.Size = 6
End With

With Chart4.Axes(owc.chAxisPositionBottom)
    .Scaling.Maximum = 1
    .Scaling.Minimum = 0
    .MajorUnit = 0.2
    .HasTitle = True
    .Title.Caption = "Expense"
    .Title.Font.Size = 6
End With

FRStakeholderComboBox.Value = ""
FRUtilityTextBox.Text = ""
FRExpenseTextBox.Text = ""

UtilityTextBox1.Text = Worksheets("Test").Cells(i, 3)
UtilityTextBox2.Text = Worksheets("Test").Cells(i, 5)
UtilityTextBox3.Text = Worksheets("Test").Cells(i, 7)
UtilityTextBox4.Text = Worksheets("Test").Cells(i, 9)
```

```vba
    ExpenseTextBox1.Text = Worksheets("Test").Cells(i, 2)
    ExpenseTextBox2.Text = Worksheets("Test").Cells(i, 4)
    ExpenseTextBox3.Text = Worksheets("Test").Cells(i, 6)
    ExpenseTextBox4.Text = Worksheets("Test").Cells(i, 8)

    Dim scenario_num As Integer
    scenario_num = Worksheets("Test").Cells(i, 1)

    i = 2

    Do Until (Worksheets("Test").Cells(i, 11) = scenario_num)
        i = i + 1
    Loop


    'fill in Attribute List
    Dim att_list As Range
    Dim numrows As Integer
    Dim Att_range As Range
    Set att_list = Worksheets("Home").Range("Attribute_list")
    Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
    numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

    Dim Array_A() As String
    ReDim Array_A(0 To numrows + 1, 0 To 2)
    Dim Array_C(20) As String
    Dim Array_D(20) As String

    Array_A(0, 0) = "Attribute Name"
    Array_A(0, 1) = "Attribute Value"

    Dim b As Integer

    b = 0

    Do While b < numrows
        Array_A(b + 1, 0) = Worksheets("Test").Cells(i, 12).Value
        Array_C(b + 1) = Worksheets("Test").Cells(i, 13).Value
        Array_D(b + 1) = Worksheets("Test").Cells(i, 14).Value
        Array_A(b + 1, 1) = Array_C(b + 1) & " " & Array_D(b + 1)
        b = b + 1
        i = i + 1
    Loop

    StakeholderAnalysis.AttributesListBox.List = Array_A
End Sub

Private Sub CommandButtonShowAll_Click()
    Load_StakeholderAnalysis
End Sub

Private Sub CommandButtonShowGraph_Click()

    'Deletes charts
    Dim ch As Chart
```

```
    For Each ch In ActiveWorkbook.Charts
        With ch
            If .Name = "Stakeholder 1" Or .Name = "Stakeholder 2" Or .Name = "Stakeholder 3" Or .Name =
"Stakeholder 4" Then
                ch.Delete
            End If
        End With
    Next

    Dim a As Integer
    Dim cellRange As String

    Select Case FRStakeholderComboBox.Value
        Case "Stakeholder 1"
            a = 2
            cellRange = "B2:C42"
        Case "Stakeholder 2"
            a = 4
            cellRange = "D2:E42"
        Case "Stakeholder 3"
            a = 6
            cellRange = "F2:G42"
        Case "Stakeholder 4"
            a = 8
            cellRange = "H2:I42"
        Case Else
            MsgBox ("Error.  Please select a stakeholder.")
    End Select

    Charts.Add

    With ActiveChart
        .ChartType = xlXYScatter
        .Name = FRStakeholderComboBox.Value

        .SetSourceData Source:=Sheets("Test").Range(cellRange), PlotBy:=x1Columns
        .HasTitle = True
        .ChartTitle.Characters.Text = FRStakeholderComboBox.Value & ": Multiple Attribute Scenario
Chart"
        .Axes(1, 1).MinimumScale = 0
        .Axes(1, 1).MaximumScale = 1
        .Axes(2, 1).MinimumScale = 0
        .Axes(2, 1).MaximumScale = 1
        .Axes(1, 1).HasTitle = True
        .Axes(1, 1).AxisTitle.Characters.Text = "Expense"
        .Axes(2, 1).HasTitle = True
        .Axes(2, 1).AxisTitle.Characters.Text = "Utility"
        .Legend.Delete
    End With

    Unload StakeholderAnalysis
    StakeholderAnalysis.Hide
End Sub

Private Sub CommandButtonUpdate_Click()
```

```vba
    ReDim ArrayAttribute(20)
    ReDim ArrayValue(20)
    ReDim ArrayUnit(20)
    Dim i As Integer
    Dim j As Integer

    Dim att_list As Range
    Dim numrows As Integer
    Dim Att_range As Range
    Set att_list = Worksheets("Home").Range("Attribute_list")
    Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
    numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

    i = 2

    Do Until (Worksheets("Test").Cells(i, 11) = Worksheets("Test").Cells(1, 14))
        i = i + 1
    Loop

    j = 0

    Do While j < numrows
        ArrayAttribute(j) = Worksheets("Test").Cells(j + i, 12).Value
        ArrayValue(j) = Worksheets("Test").Cells(j + i, 13).Value
        ArrayUnit(j) = Worksheets("Test").Cells(j + i, 14).Value
        j = j + 1
    Loop

    Dim Array_A() As String
    ReDim Array_A(0 To numrows + 1, 0 To 2)

    Array_A(0, 0) = "Attribute Name"
    Array_A(0, 1) = "Attribute Value"

    i = 0

    Do While i < numrows
        Array_A(i + 1, 0) = ArrayAttribute(i)
        Array_A(i + 1, 1) = ArrayValue(i) & " " & ArrayUnit(i)
        i = i + 1
    Loop

    AttributeScenario.AttributesListBox.List = Array_A

    Load AttributeScenario
    AttributeScenario.show
End Sub

Public Function CalcUtility(Stakeholder As String) As Double

    Dim att_list As Range
    Dim numrows As Integer
    Dim Att_range As Range
    Set att_list = Worksheets("Home").Range("Attribute_list")
    Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
```

```vba
numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

Dim Array_att(20) As String
Dim i As Integer
i = 0

Do While i < numrows
   Array_att(i) = Att_range(i + 1).Value
   i = i + 1
Loop

'this allows to find index of newest Attribute Scenario
i = 2

Do Until (Worksheets("Test").Cells(i, 11) = Worksheets("Test").Cells(1, 14))
   i = i + 1
Loop

'Finds all Ui(Xi) values
Dim UtilVal(20) As Double
Dim AttVal As Double

Dim j As Integer
Dim rise As Double
Dim run As Double
Dim yint As Double

j = 0
g = 0

Do While j < numrows
   AttVal = Worksheets("Test").Cells(j + i, 13)
   Do While AttVal > Worksheets(Array_att(j)).Cells(g + 2, 23)
      g = g + 1
   Loop
   'now at cell k, the val is great than AttVal

   If g = 0 Then
      rise = Worksheets(Array_att(j)).Cells(g + 2, 24)
      run = Worksheets(Array_att(j)).Cells(g + 2, 23)
      If run = 0 Then
         rise = 0
         run = 1
      End If
   Else
      rise = Worksheets(Array_att(j)).Cells(g + 2, 24) - Worksheets(Array_att(j)).Cells(g + 1, 24)
      run = Worksheets(Array_att(j)).Cells(g + 2, 23) - Worksheets(Array_att(j)).Cells(g + 1, 23)
   End If

   yint = Worksheets(Array_att(j)).Cells(g + 2, 24) - (rise / run) * Worksheets(Array_att(j)).Cells(g + 2, 23)
   UtilVal(j) = AttVal * rise / run + yint
   g = 0
   j = j + 1
Loop
```

```
'Finds all k corner points
Dim K(20) As Double

j = 0
g = 0

Do While j < numrows
   Do While Worksheets(Array_att(j)).Cells(g + 2, 17) <> "I"
      g = g + 1
   Loop

   K(j) = Worksheets(Array_att(j)).Cells(g + 2, 16)
   g = 0
   j = j + 1
Loop


'Find Kconstant
Dim Kconstant As Double
Dim temp As Double
Dim zeroval As Double
Dim smallest As Double
Dim smaller As Double
Dim Ksave1 As Double
Dim Ksave2 As Double

j = 0
smallest = 1000
smaller = 1000
Ksave1 = 0
Ksave2 = 0

Kconstant = -1
temp = 1

Do While Kconstant < 1.1
   Do While j < numrows
      temp = (Kconstant * K(j) + 1) * temp
      j = j + 1
   Loop

   zeroval = Abs(temp - (Kconstant + 1))

   If zeroval < smallest Then
      smaller = smallest
      Ksave2 = Ksave1

      smallest = zeroval
      Ksave1 = Kconstant
   ElseIf zeroval < smaller Then
      smaller = zeroval
      Ksave2 = Kconstant
   End If

   Kconstant = Kconstant + 0.1
```

```
      temp = 1
      j = 0
   Loop

   Dim Kfinal As Double
   j = 0
   smallest = 1000
   temp = 1
   Ksave1 = Round(Ksave1, 2)
   Ksave2 = Round(Ksave2, 2)

   Do While Ksave1 < Ksave2 + 0.01
      If Ksave1 = 0 Then
         Ksave1 = Ksave1 + 0.01
      End If

      Do While j < numrows
         temp = (Ksave1 * K(j) + 1) * temp
         j = j + 1
      Loop

      zeroval = Abs(temp - (Ksave1 + 1))

      If zeroval < smallest Then
         smallest = zeroval
         Kfinal = Ksave1
      End If

      Ksave1 = Ksave1 + 0.01
      temp = 1
      j = 0
   Loop

   Dim finalUtil As Double

   j = 0
   temp = 1

   Do While j < numrows
      temp = (Kfinal * K(j) * UtilVal(j) + 1) * temp
      j = j + 1
   Loop

   finalUtil = Round(((temp - 1) / Kfinal), 2)
   CalcUtility = finalUtil

End Function

Public Function CalcExpense(Stakeholder As String) As Double
   Dim randomnumber As Double

   randomnumber = Round((0 + Rnd * (0.15)), 2)
   CalcExpense = CalcUtility(Stakeholder) + randomnumber

End Function
```

9.3 Attribute Scenario Update Form Code

```
'Code that handles all event actions on the Attribute Scenario Form
Private Sub CommandButtonCancel_Click()
    AttributeScenario.Hide
    Unload AttributeScenario
End Sub

Private Sub CommandButtonScenarioComplete_Click()
    Dim i As Integer

    Dim att_list As Range
    Dim numrows As Integer
    Dim Att_range As Range
    Set att_list = Worksheets("Home").Range("Attribute_list")
    Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
    numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

    i = 2

    Do Until (Worksheets("Test").Cells(i, 11) = Worksheets("Test").Cells(1, 14))
        i = i + 1
    Loop

    'i is the row where the new Attribute Scenario can be put down
    i = i + numrows
    'add one more row for spacing of attribute sets
    i = i + 1

    'the empty space index is incremented
    Worksheets("Test").Cells(1, 14) = Worksheets("Test").Cells(1, 14) + 1

    Dim j As Integer

    j = 0

    Do While j < numrows
        Worksheets("Test").Cells(i, 11) = Worksheets("Test").Cells(1, 14)
        Worksheets("Test").Cells(i, 12) = ArrayAttribute(j)
        Worksheets("Test").Cells(i, 13) = ArrayValue(j)
        Worksheets("Test").Cells(i, 14) = ArrayUnit(j)
        j = j + 1
        i = i + 1
    Loop

    'number the Attribute Scenario for naming of Utility and Expenses

    Dim tmp As Integer

    tmp = Worksheets("Test").Cells(1, 14) + 1

    Worksheets("Test").Cells(tmp, 1) = Worksheets("Test").Cells(1, 14)

    Worksheets("Test").Cells(tmp, 2) = StakeholderAnalysis.CalcExpense("Stakeholder1")
    Worksheets("Test").Cells(tmp, 3) = StakeholderAnalysis.CalcUtility("Stakeholder1")
```

```
    Worksheets("Test").Cells(tmp, 4) = StakeholderAnalysis.CalcExpense("Stakeholder2")
    Worksheets("Test").Cells(tmp, 5) = StakeholderAnalysis.CalcUtility("Stakeholder2")
    Worksheets("Test").Cells(tmp, 6) = StakeholderAnalysis.CalcExpense("Stakeholder3")
    Worksheets("Test").Cells(tmp, 7) = StakeholderAnalysis.CalcUtility("Stakeholder3")
    Worksheets("Test").Cells(tmp, 8) = StakeholderAnalysis.CalcExpense("Stakeholder4")
    Worksheets("Test").Cells(tmp, 9) = StakeholderAnalysis.CalcUtility("Stakeholder4")

    Load_StakeholderAnalysis

    AttributeScenario.Hide
    Unload AttributeScenario
End Sub

Private Sub CommandButtonUpdateAttribute_Click()

    Dim i As Integer

    i = 0

    Do Until (ArrayAttribute(i) = AttributesListBox.Value)
       i = i + 1
    Loop


    UpdateAttr.PromptLabel.Caption = "Current value for " & ArrayAttribute(i) & " is " & ArrayValue(i) &
" " & ArrayUnit(i)
    UpdateAttr.Caption = ArrayAttribute(i)
    UpdateAttr.MinMaxLabel.Caption = "(Min value, Max value) : (" & Find_Low(ArrayAttribute(i)) & ", "
& Find_High(ArrayAttribute(i)) & ")"

    Load UpdateAttr
    UpdateAttr.show
End Sub
```

9.4 Update Attribute Form Code

```
'Code that handles all event actions on the Update Attribute Form
Private Sub CommandButtonCancel_Click()
    UpdateAttr.Hide
    Unload UpdateAttr
End Sub

Private Sub CommandButtonUpdate_Click()
    Dim i As Integer

    i = 0

    Do Until (ArrayAttribute(i) = UpdateAttr.Caption)
       i = i + 1
    Loop

    ArrayValue(i) = TextBoxAttributeValue.Text
    ArrayUnit(i) = TextBoxAttributeUnit.Text

    UpdateAttributeScenario
```

```vba
      UpdateAttr.Hide
      Unload UpdateAttr
End Sub

Sub UpdateAttributeScenario()

   Dim i As Integer

   Dim att_list As Range
   Dim numrows As Integer
   Dim Att_range As Range
   Set att_list = Worksheets("Home").Range("Attribute_list")
   Set Att_range = ActiveWorkbook.Names("Attribute_list").RefersToRange
   numrows = Att_range.Rows.Count 'numrows contains the number of Attributes

   Dim Array_A() As String
   ReDim Array_A(0 To numrows + 1, 0 To 2)

   Array_A(0, 0) = "Attribute Name"
   Array_A(0, 1) = "Attribute Value"

   i = 0

   Do While i < numrows
      Array_A(i + 1, 0) = ArrayAttribute(i)
      Array_A(i + 1, 1) = ArrayValue(i) & " " & ArrayUnit(i)
      i = i + 1
   Loop

   AttributeScenario.AttributesListBox.List = Array_A

   Load AttributeScenario
End Sub
```

9.5 Find Relations Code

```vba
'Code that handles the Find Relations functionality
Sub Find_Relations(Stakeholder As String, Utility As String, Expense As String)

   Dim incr As Integer
   Dim attr As Integer
   Dim i As Integer

   Select Case Stakeholder
      Case "Stakeholder 1"
         incr = 2
      Case "Stakeholder 2"
         incr = 4
      Case "Stakeholder 3"
         incr = 6
      Case "Stakeholder 4"
         incr = 8
   End Select
```

```
    attr = 0

    For i = 1 To 40
        If (Utility = Worksheets("Test").Cells(i + 1, incr + 1)) And (Expense = Worksheets("Test").Cells(i +
1, incr)) Then
            attr = Worksheets("Test").Cells(i + 1, 1)
        End If
    Next

    If attr = 0 Then
        MsgBox "The entered Utility and Expense Values do not correlate to an Attribute set for the specified
Stakeholder"
    ElseIf attr <> 0 Then
    'display relevant numbers
    End If

End Sub
```