Air Force Institute of Technology

AFIT Scholar

Theses and Dissertations

Student Graduate Works

3-2008

Mission Impact Analysis Visualization for Enhanced Situational Awareness

Sean C. M. Carroll

Follow this and additional works at: https://scholar.afit.edu/etd

Part of the Computer Engineering Commons

Recommended Citation

Carroll, Sean C. M., "Mission Impact Analysis Visualization for Enhanced Situational Awareness" (2008). *Theses and Dissertations*. 2737.
https://scholar.afit.edu/etd/2737

This Thesis is brought to you for free and open access by the Student Graduate Works at AFIT Scholar. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of AFIT Scholar. For more information, please contact AFIT.ENWL.Repository@us.af.mil.

Mission Impact Analysis
Visualization For
Enhanced Situational Awareness

THESIS

Sean C. M. Carroll, Captain, USAF

AFIT/GCO/ENG/08-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GCO/ENG/08-01

Mission Impact Analysis
Visualization For
Enhanced Situational Awareness

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science (Cyber Operations)

Sean C. M. Carroll, M.S.I.T.M.

Captain, USAF

March 2008

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GCO/ENG/08-01

Mission Impact Analysis Visualization For Enhanced Situational Awareness

Sean C. M. Carroll, M.S.I.T.M.

Captain, USAF

Approved:

LtCol S.H. Kurkowski, PhD (Chairman)	26 Feb 08 date
Maj P.D. Williams, PhD (Member)	26 Feb 08 date
Capt T.R. Andel, PhD (Member)	26 FEB 08 date

AFIT/GCO/ENG/08-01

Abstract

This research effort examines the creation of mission impact analysis visualizations to enhance situational awareness. It focuses on using `prefuse` to create a visualization that allows the user to quickly understand the impact of the failure of any element needed directly or indirectly for a mission. The visualization correctly identifies the direct or indirect impact on physical requirements such as network links and servers as well as non-physical elements such as the generation of a report, or ability to perform a task. The visualization provides an overview of the situation, as well as including enhancements to allow for greater detail on any element to be viewed. The result of this research is the foundation for a tool to allow commanders and others, at a glance, to understand the scope of mission impact when an outage occurs.

Acknowledgements

I would like to express my sincere appreciation to my faculty advisor, LtCol Kurkowski, for his guidance and support throughout the course of this thesis effort. The insight and experience was certainly appreciated. I would also like to thank my committee members for their insight and guidance. Last, but certainly not least, I would like to thank my family for all of their support and sacrifices that enabled me to complete my research and thesis.

Sean C. M. Carroll

Table of Contents

Page

List of Figures

Figure Page

List of Tables

List of Abbreviations

Mission Impact Analysis Visualization For Enhanced Situational Awareness

I. Introduction

In today's complex military environment, the demand by and on commanders for accurate, up-to-date information is critical. The success or failure of any particular mission is affected by an increasing task diversity, and the inter-dependence of various functions relating to that mission. This creates an environment where a network of functions, their individual impact, and their aggregate become an integral aspect of every military operation. As a result, network and system failures at any level of the mission process impact the commander, the command decisions, and the mission outcome. Additionally, the inter-dependence of these various functions become interwoven to the degree that the failure of a single component of the network can impact the remaining infrastructure and the missions dependent upon that infrastructure. Determining the degree of that impact in a timely manner is of critical importance to command decisions.

In the past, the communication community strived to show the commander a picture of the physical network along with the impact on the network of individual failures. Unfortunately, this physical representation is often insufficient for the needs of a commander. In order to make informed decisions, it is important for a commander to have timely, dependable information, as the problem relates to three distinct areas.

- First, what is the impact of the failures at the equipment level, in other words, the effect of individual equipment or services that fail? Examples of physical equipment failures would include routers, network links, and servers. E-mail and verification servers are examples of services that might fail.

- Second, what is the full impact that the original failure has had, in causing a "cascade" of failures to other related functions? One example of a cascading failure is demonstrated in looking at the failure of the verification server. Network files and E-mail also have effectively failed, since attempts to access them can no longer be authenticated.
- Third, what is the cumulative impact, beyond just the equipment, of these failures on the specific mission identified?

In a crisis situation each of these areas must be addressed quickly and accurately. Even though the network may be dispersed and distributed over a large geographical area and/or over several organizations. A dependable centralized information source is necessary. Access to this source ensures critical data can be obtained, avoiding the need to contact multiple individuals or offices, before even a preliminary impact report can be generated.

By utilizing commercial and internal network monitoring software, it is possible for a Network Control Center (NCC) to construct a graphical representation of the entire network including links and servers. This representation can be used to identify failed equipment, as well as the equipment and services dependent on it. There are current commercial network management solutions that do this. One such example is "Whats Up Gold", as can be seen in Figures 1.1, 1.2, and 1.3. "Whats Up Gold" can be used to identify if a server, computer, service, or link is functioning.

The software can even be loaded with hardware dependencies to cover the cascading effect of hardware failures as well as location data to indicate where the equipment is housed. However, it provides no information on what will happen beyond the hardware and server level if something fails. This is left to the personnel monitoring the software to determine. Such visualization does address the first two areas of concern, i.e. identifying the failure of the individual equipment or server service shown in Figure 1.1, and identifying the impact of that failure's cascade effect on other inter-dependent equipment or server services shown in Figure 1.2.

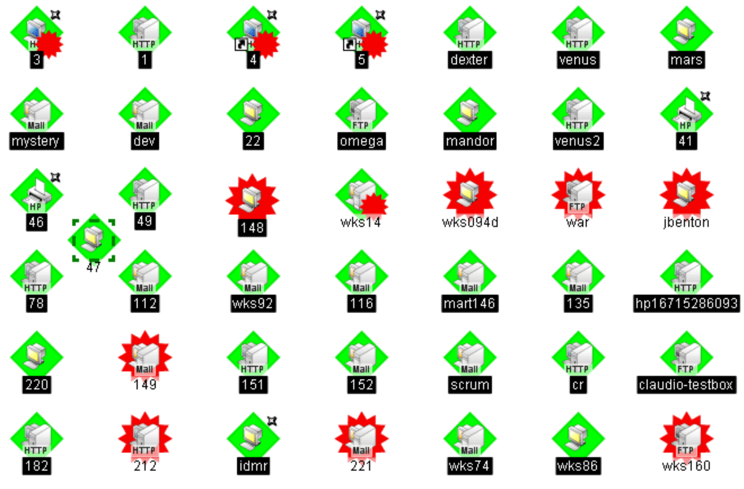

Figure 1.1: This screen shot demonstrates the basic ability of Whats Up Gold to monitor the status of network hardware and server services. [2].

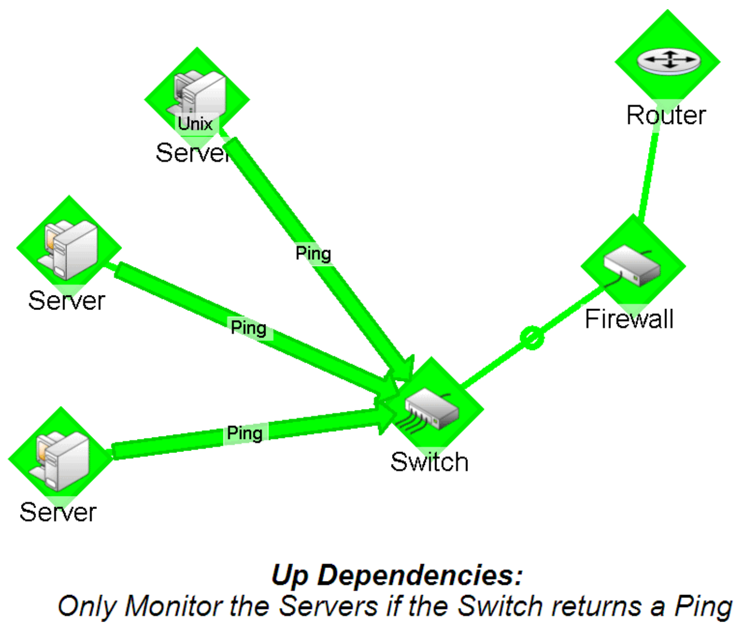

Figure 1.2: This screen shot demonstrates the basic ability of Whats Up Gold to monitor dependencies within the network of hardware and server services [2].

The framework utilized in "Whats Up Gold" and similar visualizations attempt to address the third requirement by allowing the creation of location maps, shown in Figure 1.3. These maps do not contain enough information to accurately identify all affected parties, just the location of affected hardware. It also does not have any

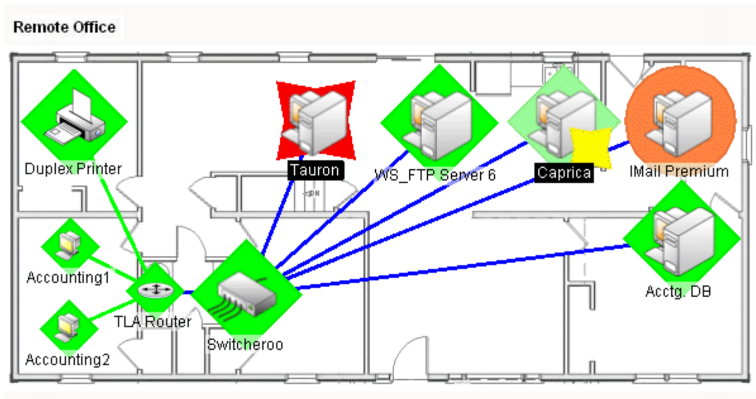

Figure 1.3: This screen shot demonstrates the ability for users of Whats Up Gold to create a graphical representation of equipment layout. This can then be used by individuals to guess at who will be affected if equipment fails [2].

way of defining non-physical elements in the layout. The software can be used as a baseline to develop a visualization that will determine the cumulative impact on a specific mission plan and other tasks that exist beyond the hardware level.

The following scenario, composed of three situations, illustrates how even with this data the NCC is unable to provide all of the information a commander needs. In each case the network maps are up to date, the wiring diagrams are accurate, and the location of the hardware is pinpointed. Yet, due to the lack of knowledge as to what the equipment is used for and how that in turn affects the mission, failure is only averted by heroic acts and fast thinking instead of prevented through good planning and organization.

1.1 Scenario

This scenario consists of a commander of a typical moderate sized Air Force (AF) base with warehouses, hangers, a flight line, office buildings, dining facility, dorms, a theater and various small buildings ranging from communication and radar shelters to guard shacks. The base housing and a large percentage of the communication infrastructure and maintenance is run by general schedule (GS) employees and civilian contractors, as are a substantial percentage of other functions on base. The flight line has a moderate to high operational tempo. All base personnel are in final preparations for an Operational Readiness Inspection (ORI).

1.1.1 Situation One: Prior to the arrival of the inspectors, a power spike causes an old uninterrupted power supply (UPS) in the base theater communications closet to fail. This particular UPS is only used to provide power to the network router which provides connectivity to the base theater. The night shift in the NCC notices the router outage and verifies, via the network wiring diagram, that the router in question is not listed as mission critical and only services the theater. It is then added to the list of projects for the day shift to address. Upon arrival, the day shift inspects the router, identifies the faulty UPS, and orders a new replacement unit since the router is not listed as "mission critical". At around the time the day shift is going off duty, the Commander and others enter the theater for the final walk-through prior to the ORI in brief the next day. During this walk-through, it was discovered the Star Spangled Banner footage was not available for display as the theater no longer had network connectivity. The projectionist then alerts the NCC that the Star Spangled Banner video footage used by the theater is no longer kept at the theater. Instead, it is now a digital file on the network file server and from there, it is streamed to the screens for display.

As a consequence of the incomplete information the NCC possessed and the fact that the NCC is unaware of what units use the network several consequences follow. Not only was the commander involved in an incident that could have been resolved the night before, but a scramble by NCC personnel would now take place to find a replacement UPS, in order for the walk through to be completed and the theater made ready for the ORI team. If the cascading failures beyond the hardware had been identified in the visualization the NCC used, then the resulting consequences would have been very different. Not only could the crisis have been handled without commander involvement, it would not have developed into a crisis at all. Instead it would simply have been an everyday problem quickly solved at the lowest level. In this situation to the best of the NCC's knowledge there were apparently only three relevant items.

- The UPS
- The router
- The Ethernet port on the wall

In reality, the chain of dependency and list of relevant items was much larger. A more accurate list encompasses elements beyond that of hardware and includes things such as mission tasks and files, as well as the elements needed to access or complete these items.

- The Star Spangled Banner file
- The file server
- The router the file server connects to
- The link to the buildings router
- The UPS
- The buildings router
- The Ethernet port on the wall.
- The Air Force mandated task to play the Star Spangled Banner

If the NCC was aware of the requirement to access a file on the file server they would have used other information at their disposal to learn of the about items on this list. This in turn would have given them the a more accurate picture of the situation, but still would have required personnel to put the pieces together and correlate the information.

1.1.2 Situation Two: At 0600 on the second day of the ORI, the NCC detects a failure of Router 0375x0122c. Based on their naming convention, this means the third router in communications closet 122 of building 375 has gone down. The technician who first notices the problem opens a trouble ticket (A1123) and begins to use the network wiring diagram to trace the hardware dependent on this router to ascertain the impact of this outage. While the other two routers in that communications closet are at capacity, router 0375x122c is only providing connectivity to two jacks in room 102 of building 375. By 0630, the technician has managed to reach the building manager. The building manager states he believes room 102 is the tool storage area for radar maintenance, but promises to call back once this is verified. At 0645 the building manager calls back verifying the room is not an office but is used for tool storage. The building manager promises to notify the Chief of Maintenance at the 0800 daily meeting. The technician marks trouble ticket A1123 for routine repairs, once the day shift arrives. At 0730, during NCC shift change-over, a frantic young airman calls the NCC Help Line because two of their office computers have lost network connectivity. The technician tries to determine if any patches were pushed out by the NCC, as the ORI inspectors were just in the shop to rate the Preventive Maintenance Inspection (PMI) procedures. After some investigation, it is determined these two computers are physically located in room 102 and used by the radar maintenance shop to track tool inventory, per AFI. Furthermore, the inventory database is not actually housed on the machines in room 102; these computers are simply client machines that must connect to a networked server. Trouble ticket A1123 is elevated to a Priority 1 and a team is dispatched immediately to restore connectivity to the tool room. The individuals being inspected then check out their tools and get to the air field to perform the PMI, 30 minutes behind schedule.

Once again, this situation developed into a crisis because the NCC did not have the scope of information needed to do the job that is expected of them. They had all the information that they were supposed to have and performed every action that policy and good judgment would require, but it still was not enough and as a result the NCC personnel would most likely be blamed for the crisis. This is a direct result of the NCC only being aware of a portion of the elements involved in the situation. Specifically the following:

- Router
- The Ethernet port on the wall in room 102

The chain of dependency and list of relevant items was once again much longer. A more accurate list encompasses elements beyond that of hardware and includes things such as mission tasks and files as well as the elements needed to acquire the direct requirements.

- Tools database
- The database server
- The router the server connects to
- The link to the building's router
- The building's router.
- The router for room 102
- The Ethernet port on the wall of room 102.
- The computers in room 102.
- The mission requirement to use a database to track inventory and check out tools

As in the previous situation, had the NCC been aware of the mission requirement, to use a remote database to track tools, the trouble ticket A1123 would have been flagged as a priority 1 the night before, and a fix or workaround would have been in place prior to anyone arriving for duty. The crisis would have been circumnavigated and simply been an entry in the night shift's log.

1.1.3 Situation Three:

At 1400 on the third day of the ORI, the front gate guards receive an exercise input; "A driver loses control of their vehicle. It swerves off the road and drives through building 74, effectively destroying it". Within minutes, the commander is aware of the situation and has personnel trying to determine who controls that building and who is impacted by its destruction. Personnel in every squadron scramble through binders and records, attempting to determine who controls that building and what it is used for. It is identified as a communications building; however, the Communications Squadron is not responsible for it. Records show that the base leases data services from an outside company, and that company is listed as the POC for the building. While phone calls are being made to the company, individuals in the Communications Squadron are now going over lists of critical equipment and facilities, such as the radar dishes and shelters, in an attempt to determine what impact the loss of this building will have. After an exhaustive search, four entries in the document are found to reference building 74, all four of these are defined as links. Specifically, the data links 12923 and 57352 and the phone links 16923 and 20363. The NCC, using the network and phone wiring diagrams, trace down what those four links are used for, and discover that the affected data and phone links are the primary and backup trunks coming into and out from the base. Without them, the base has no outside phone or Internet connectivity.

In this situation the NCC had all of the relevant facts but they were not organized and visualized in a manner to allow quick access to them. Based on initial reports there was only a single element in play, that being building 74. When in reality there were five key elements, the building and the four trunks, with a cascading impact affecting the entire base's communications network. Yet, since the building and trunks were under the purview of a contractor, the NCC was originally blind to the significance of the situation and valuable time was lost while they tried to contact the contractor and pored through documents trying to determine the significance of the building.

1.2 Scenario Analysis

In these three situations the NCC did not have instant access to the complete information needed to make the correct decisions or to pass on to the commander so that an informed decision could be made. All of the information needed was available via experts that the NCC could reach during normal operation hours. Though this process would seem sufficient at first glance, when considered in the light of these scenarios, the failure of the NCC to have immediate access to the entire scope of information is shown to be detrimental to operational effectiveness. Currently, no system traces the impact of equipment failures past its direct impact on other equipment and server services, or addresses the cascade effect of such failures to the missions or tasks above the machine and hardware level. This demonstrates the need for a system with the ability to trace the physical network equipment up through all of the missions and mission tasks that rely on the equipment, as well as to the missions that rely on the mission tasks, then eventually to trace the impact of those mission failures on other missions. Furthermore, this system must present the information in a format that can be easily understood in a timely manner.

If the NCC had a system in place that captured all of the information directly and indirectly at their disposal, the decisions the NCC makes and the priorities they set would be much more informed. Information would include data concerning who needs what equipment, network services, etc., as well as why they need it, along with what that equipment also needs. Furthermore, if there was a system to visualize this interconnection of needs and dependencies the decision process would be significantly streamlined and accelerated.

1.3 Data Collection

Before any system can be designed to show the impact of equipment failure on the mission, two sets of data must be captured: operational status and relationships or dependencies. The operational status of the equipment must be collected as one of the datasets. Preferably this data would be collected using an automated tie-in, such as cybercraft, or existing network monitoring software. Another possible solution would involve a manual process with an operator annotating a failure that has been detected. This situation would result in a delay of the analysis but the overall result would still be faster than attempting to determine the scope of affected systems and missions without a visualization tool. The relationships and dependencies between the equipment and the mission is more difficult to capture or define, but could be captured the same way the Air Force generates Mission Essential Task Lists (METL), reports on single points of failure, and equipment lists. Other possibilities for collecting this data are presented in recent research [13, 14].

1.4 Conveying the Information

The method chosen to convey the information is a critical component in any system. As the scenario's situations demonstrate, simply having access to the information is not enough. It must be stored and presented in such a way as to allow individuals to quickly and accurately assess a situation. This makes a visual representation much more desirable, as opposed to a written report. Visualization allows individuals to determine, at a glance, the level of impact from an event, quickly differentiating the situation that only affects a single office from one affecting large sections of the base.

This visualization would not necessarily need to display the dependencies between elements in a way that would allow a human to quickly trace what is affected. Instead, it could be set up so that the computer keeps track of the dependencies and in some manner brings the affected elements to the attention of the personnel. This would simplify many of the difficulties facing attempts to create a visualization, such as relating the physical location of equipment to a mission or task that takes place on the other side of base. Or the difficulty of finding or developing a visualization format to make clear which elements are physical, such as routers and wiring, and non-physical elements, for instance tasks, reports, or software.

This research seeks to demonstrate that a visualization incorporating automated mission impact analysis to generate an accurate overview is possible, thus greatly enhancing the situational awareness of the NCC and commander. To this end it is necessary to begin by examining other efforts and work that has been completed in the area of mission impact analysis, as well as research in visualization of such data to form a starting place for the research at hand, contained in chapter two. Chapter Three addresses the methodology utilized for generating the visualization capable of providing enhanced situational awareness. Chapter Four will address the outcome, while Chapter Five will provide conclusions drawn from this research and highlight its potential impact.

In other words, I hypothesis that it by looking at the the problem in a different manner that many current researchers it is possible to create a usable mission impact visualization for enhanced situational awareness using current technology that will meet the minimum needs of the United States Air Force. The to prove this I will constructed such a visualization.

II. Background Information

This chapter describes some issues critical for an understanding of the problem and the development of a solution. Specifically, it covers the meaning of "mission impact assessment" as it pertains to this research. Next, is a brief overview of issues related to the problem of creating an automated mission impact analysis representation. Finally, the elements needed for a working solution are discussed.

2.1 Mission Impact Assessment

In order to provide increased situational awareness, mission impact assessment is necessary. This problem breaks down into two segments. The first is data acquisition and the second is visualization of that data. Though this research focuses more on the visualization aspect of the problem, it is important to keep both aspects of the problem in mind and plan accordingly. Without data the visualization would be useless. In the case of automated mission impact analysis the visualization must aggregate information and be easy to use and understand.

For the purposes of this research the term "mission impact analysis" is used in reference to the identification of individual and/or cascading failures that can be caused by equipment, system, or other failures. These other failures could include the destruction of a building, a generator, or a mission task that is not accomplished. In other words the mission impact analysis of the failure of any identified requirement would be a list of the resulting failures, or possible failures. These failures and potential failures could be equipment failures, mission tasks, or anything else affected directly or indirectly as a result from the primary failure. This assessment could be accomplished by a human, by a computer, or by some combination of the two.

For example take a situation with the following elements and needs, depicted in Table 2.1. The situation begins when link A is cut by a road repair crew. This link in turn provided connectivity for router B which in turn provides connectivity for building C. It is at this point that most network diagrams would cease to be helpful. In order to provide a useful mission impact analysis the chain of events would need to

Table 2.1: Elements used in describing a theoretical cascading failure beginning with a cut communication link.

Label	Description	Requirements
A	a communications Link	
B	a router	A
C	a building	C
D	a computer	D
E	a task	D
F	a mission	E
G	another mission	E
H	a third mission	E
I	a forth mission	E

be followed further, until the ramification on the mission (or missions) are revealed. In this example building C contains a computer which we shall refer to as D. Computer D is used to accomplish task E. Task E is an aspect of scheduling passengers and cargo for air missions, specifically identifying passenger and cargo cancellations to determine if any individuals or cargo on the wait list can be loaded. Task E is considered a critical task necessary for successful planning and execution of air transport missions, as such the air transport missions F, G, H, and I are now in danger of failing. These failures could cascade and cause failures on other bases. Such failures may mean something as simple as a soldier deploying to, or returning from, Iraq does not get his luggage. Or it may be something much more life endangering. For example, cryptographic keys being delayed so that units are not able to communicate in a secure manner. This failure could force commanders to choose between risking lives by using unsecure communications, or accepting mission failures by refusing to take actions and execute missions until the cryptographic keys arrive.

The mission impact analysis, as defined for this research, identifies that the failure of link A may cause B, C, D, E, F, G, H, and I to fail. However, this phase of the research will not identify solutions to avoid these failures or take into account non-standard methods personnel may take in order to avoid mission failure.

2.2 Data Acquisition

Developing a procedure for the acquisition of the necessary data consists of two stages. The first stage is to determine if the all, or some, of data is already being acquired. The second stage is to determine at least one method capable of acquiring any data not already being collected and to ensure it is usable. Once it can be determined that the data can in fact be successfully collected through some method focus can be given to the visualization.

2.2.1 Current Acquisition Method. Most organizations have some method for acquiring information pertaining to what is required to accomplish a task. Current US Air Force policies and procedures require the creation of Mission Essential Task Lists and/or Critical Equipment Lists. One of the intents behind the creation of these lists is to allow individuals to assess the impact on the mission if an element on the lists fails. However, the lists do not always provide useful information, since they are often split up between offices and normally take into account only first order requirements and needs. The list may include a requirement for access to a server, but does not take into account that the server requires access to the Internet. Consequently, the final solution used in conjunction with the visualization must allow and/or encourage the integration and immediate access to such data.

2.2.2 Possible Future Acquisition Methods. "A Survey of Active Network Research" by D. L. Tennenhouse, et al in 1997, provided an excellent overview of the concepts proposed in active network designs [10]. This research is important in terms of data collection because one of the proposed abilities of an active network is that the network itself can determine the interdependencies of not only the hardware, but of missions and elements beyond the hardware level by monitoring and tracking what information is sent by who or what and where that information is sent. Active network research is characterized by the statement "nodes can perform computations on and modification of packets." With nodes being any computational device a packet passes through such as a router, switch, or computer. In order to accomplish these computations and modifications, two primary branches of active network research are being pursued: discrete and integrated active networks. For the discrete approach, a packet is received by the router or switch and the header information is read and appropriate changes are made based on programs already coded and waiting in the device. Currently, both routers and switches already take in the packet, then based on header information execute pre-stored programs, specifically the routing of that packet to the appropriate port. As such, the discrete approach can be seen as an extension of the functions that make up the routers and switches.

The integrated approach takes this concept one step further. Instead of the router or switch making these computations based on pre-stored programs, every packet contains the program to be executed. The router or switch compiles and/or executes this program. This added flexibility comes at the cost of a more complex router and switch, since it must be able to compile and run new programs on the fly instead of only running programs it has been optimized for. While active networks that utilize the integrated approach are potentially more flexible then networks that utilize the discrete approach, both have the potential to collect a vast amount of useful data. They could track who communicates with whom and what information is being sent, identify who receives certain reports, what servers and systems are dependent on others, or identify if a piece of information is reaching an office by multiple routes.

In short, an active network has the potential of collecting more statistical and informational data on what and how information travels over the network then actually travels over the network. Thus the flexibility of both of these approaches and the potential data available for collection make an active network tailor made for automated systems for mission impact analysis, since they could be set up to collect or formulate virtually any information needed for the analysis. It is possible an active network could, or would include the capability to perform impact analysis without the need for additional programs.

J. E. Stanley's thesis, "Enabling Network Centric Warfare Through Operational Impact Analysis Automation" [14] offers promising solutions to automated impact analysis, and demonstrates that by automating network analysis via an active network design it is possible to capture a great deal of information. This information can be leveraged to not only optimize a network but to also correlate the impact of network activity and outages on missions and mission objectives. This approach offers a great deal of flexibility and tools for network configuration, network management, and network monitoring. Once active networks are a reality and the future works proposed in this research are realized, a system similar to the one proposed would be invaluable for collecting the data needed for a mission impact analysis visualization.

Alfred K. Shaw, takes things a step further and proposes a model for determining the relationship between the various tasks and elements needed to accomplish a mission impact analysis in his thesis "A Model for Performing Mission Impact Analysis of Network Outages" [13]. This model does not require an active network to gather the information correlating how various elements are interdependent and affect the mission. Instead, he proposes a methodology that can be followed by expert human analysts that will result in a 100% complete and accurate model of the dependencies for a mission or task.

Another solution for gathering the data set representing the dependencies can be accomplished without the use of active networks or expert analysts is to make use of the individuals responsible for the operation of equipment or the accomplishment of tasks. This would not necessarily yield a complete model, but it would capture the local domain knowledge and information currently in existence. In this case, the requirements and the dependencies would be generated by instructing individuals responsible for missions and tasks, to generate them and keep track of what they need to accomplish the missions or tasks assigned to them. In the case of equipment, individuals would be instructed to generate and track what is needed for their equipment to function and accomplish the tasks associated with it. While not a perfect solution, this solution has the advantage of being able to be implemented immediately, not requiring all of the Air Force's network architecture to be replaced, or hiring an expert to model the dependencies for every mission and then have these models reevaluated every time the Air Force changes how something is done.

2.3 Visualization Solution Requirements

With the data representing the dependency chain acquired, the focus shifts to finding a way to represent these dependencies in a manner that can be quickly and accurately interpreted. This research focuses on utilizing a graphical visualization for this task. Any proposed visualization must satisfy several requirements for this problem. First, it must support the needs of the United States Air Force (USAF). Second, it must make the impact analysis easier to comprehend, not more difficult or time consuming in situations where an outage occurs. Finally, scalability must be considered. Scalability is composed of two aspects, the complexity of the visualization and the amount of data the visualization is capable of representing. A solution that limits either of these aspects to a size and complexity that would only support a flight or squadron would not be beneficial.

To satisfy the needs of the Air Force, the solution must correspond to the current USAF structure and align with the Department of Defense (DoD) goals for Network Centric Warfare (NCW). The "Report on Network Centric Warfare Sense of the Report" by General Money presented to congress in 2001 provides critical information concerning the goals and objectives of NCW [11]. As does "Network Centric Warfare: An Emerging Theory" by John J. Garstka published in 2000 and the 1999 text "NETWORK CENTRIC WARFARE: Developing and Leveraging Information Superiority" [5, 16]. Simply put, while any solution for a problem may be of academic interest, if it runs contrary to the long term goals and needs of the US military it is not feasible as a real world solution.

To acquire and maintain good situational awareness, all of the details about every element of the network does not need to be available at a glance. However, outages must be easy to identify and the scope of the outage must be readily apparent. If this is accomplished the overall impact of the situation and outage can be easily understood and comprehension of the scope of the problem readily obtained. Details can then be acquired in response to the source of the outage and the missions impacted.

Scalability must be considered from the start. For example, Travis Air Force Base has over 10,000 personnel, Wright Patterson has over 13,000 personnel, and Tinker has more than 23,000 personnel. If you assume only 80 percent of these individuals have computers and of these only 50 percent require these to accomplish their missions, the resulting numbers are more than 4000 for Travis, 5200 for Wright Patterson and 9200 for Tinker. This means that at Travis Air Force Base, the smallest of these three examples, the visualization must simultaneously handle a bare minimum of 4000 elements. Once you take into account the network links, the various routers, servers, and switches, as well as the missions themselves, this number quickly grows much larger. As a result, any solution needs to have the potential to handle several thousand elements at a time.

2.4 Current Visualization Techniques

This research focuses on two dimensional visualizations. This focus is for three reasons. First, the tools for creating two dimensional visualizations are mature. Second, these types of visualization are more intuitive to create and display. Third, the majority of the graphs and visual representations used in displaying information for Air Force personnel, such as wiring diagrams, command structures, and various stop-light charts are two dimensional in nature. This has the added benefit of reducing the risk that the visualization will need to be rotated or manipulated in order to ensure a critical piece of data is not obscured.

2.4.1 Node-Link Diagram Representations. There are three basic node-link visualizations. Each of these is made up of a node or vertex and the link or edge. This allows for a very clear and concise visualization of the information elements. Each element becomes a node. The links in turn represent how these elements relate to each other.

2.4.1.1 Graphs. While all node-link representations are technically graphs, in this particular case the term graph refers to a system of nodes and links that have no limitation on how they are connected. The links have no direction associated with them. It may also be possible to traverse all or part of the graph in such a way as to arrive back at the node you started from without ever traveling back over the same node or link a second time. Figure 2.1 is an example of such a representation.

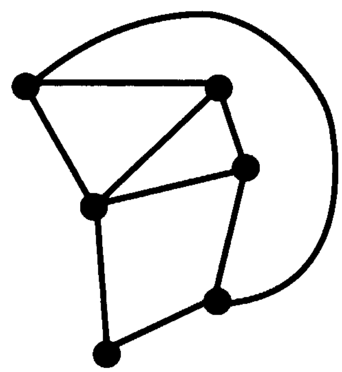

Figure 2.1: Example of a graph using node-link representation without direction associated with the links, consisting of 7 nodes and 9 links

2.4.1.2 Directed Graphs. Directed graphs are identical to graphs except they have the constraint that each link in the graph must have at least one direction associated with it. This allows for the structure of the graph to display more information. For example, not only representing that two nodes are associated with each other, but how they are associated. Figure 2.2 is an example of such a representation.

2.4.1.3 Trees. Trees contain information within the structure of the graph concerning how nodes are associated with each other. In the case of trees this information is not represented by adding directionality to the links, but instead by

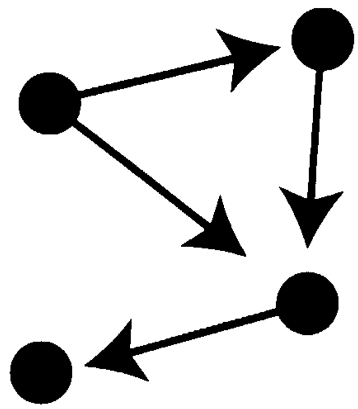

Figure 2.2: Example of a directed graph consisting of 4 nodes and 4 links

the placement of one node in relation to other nodes. Figure 2.5 is an example of such a representation.

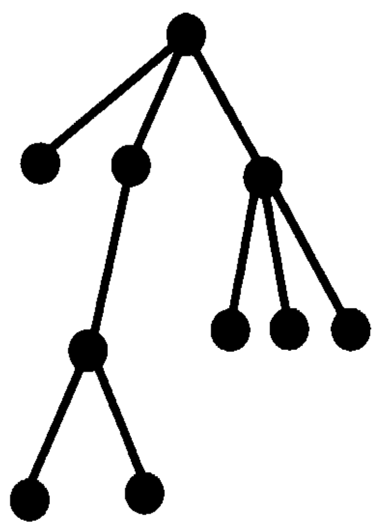

Figure 2.3: Example of a tree graph consisting of 10 nodes and 9 links

2.4.2 Other Representations. There are a variety of other visual representations that convey relationships between elements that are more complex than basic node-link representation. Some of these representations use an underlying node-link structure to store the data used in the visualization and others augment node-link representations. Examples include flow maps, tree maps, radial representations, and fish-eye representations.

2.4.2.1 Flow Maps. Flow maps work as excellent visual representations of the movement of something from one location to another (See Figure 2.4), even if the element moving is not physical in nature such as the flow of information over a computer network. The paper "Flow Map Layout" provides an excellent demonstration of this concept using three examples [17]. Figure 2.4 (a) shows the export of wine from France. Figures 2.4 (b) and 2.4 (c) illustrate the migration of individuals from California to other places in the country.

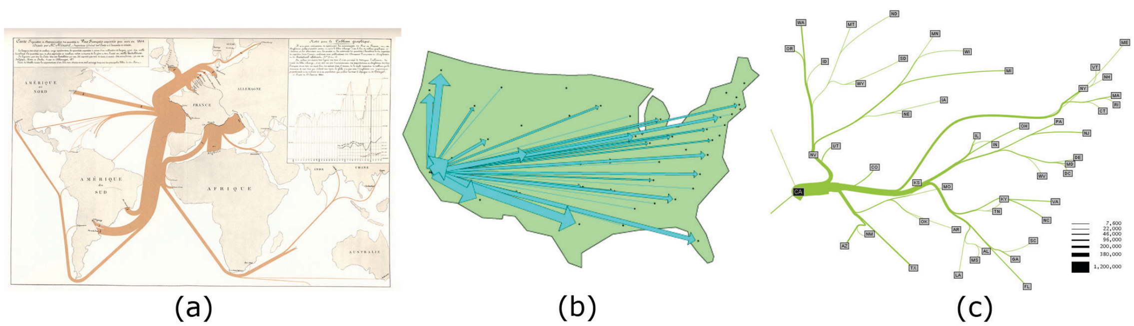

Figure 2.4: Flow Maps. (a) Minard's 1864 flow map of wine exports from France (b) and (c) show migration from California from 1995 - 2000.

2.4.2.2 Tree Maps. Tree maps represent in their structure the exact same information as trees. However, instead of displaying this information in a node-link format it uses layers to demonstrate the relationships. The paper "Flow Map Layout" has a good graphical representation of this though not all nodes are labeled [17], shown in Figure 2.5.

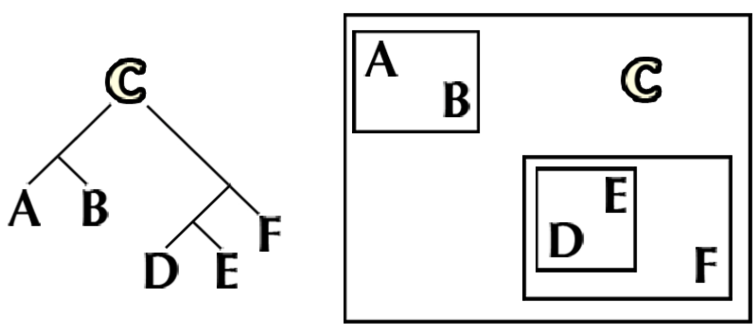

Figure 2.5: Example of a tree map and its corresponding tree graph representation

2.4.2.3 Radial Representations. Radial representations are most often seen associated with trees though they can also be used with any type of node-link representation. These are designed to make more efficient use of space as well as show everything in relation to a selected node. An example of a radial graph representing a node-link representation can be found in the paper "Animated Exploration of Dynamic Graphs with Radial Layout" [7], which is depicted in Figure 2.6.

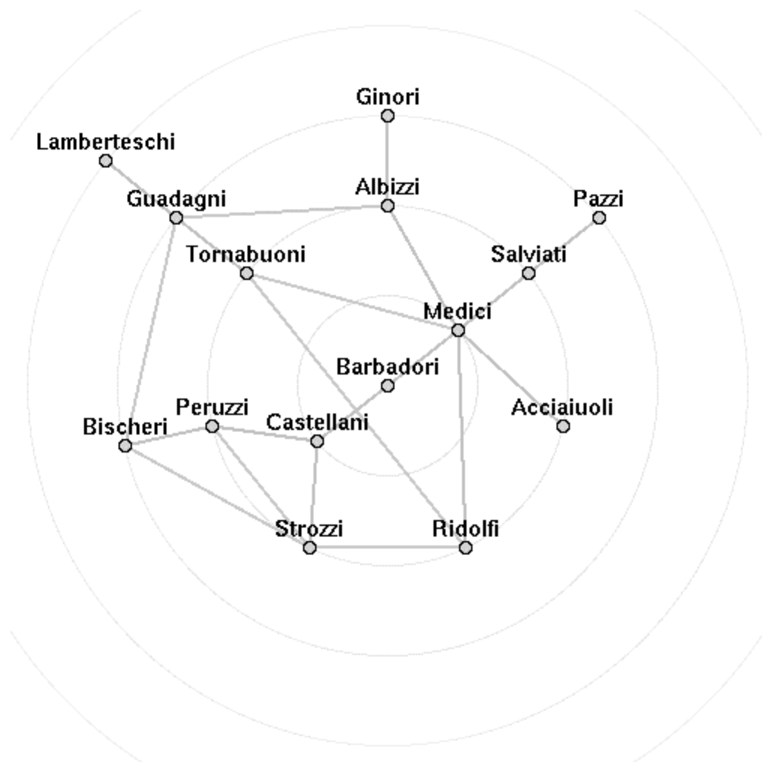

Figure 2.6: Example of radial representation based on a node-link graph with 15 labeled nodes and 20 non-directed links

2.4.2.4 Fish-Eye Representation. A fish-eye representation can be applied to any other representation. The basic premise behind this modification is that the user selects an element of the visualization to focus on. The representation is then distorted via a gradient zoom, with the greatest amount of zoom applied to the focus element and an ever decreasing amount of zoom to surrounding elements based on their relation to the focus element. As a result, elements furthest away from the focus may be unreadable. An example is shown in Figure 2.7 from "Generalized Fisheye Views of Graphs" [8].

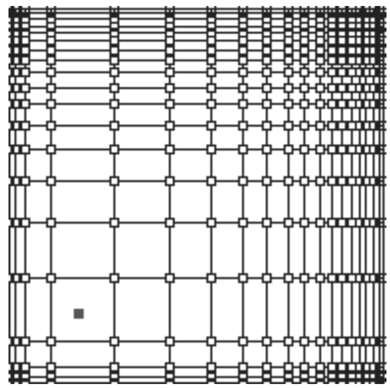

Figure 2.7: Example of fish-eye representation based on a node-link graph

2.4.3 Summary of Visualization Techniques. Each of these techniques seems suitable as they are represented. However, once the type of data and the interdependencies of that data is taken into account complications develop with some of them. All types of trees fall into this category since, in a real world system an element or node may be needed by more than one other element or node. For example, imagine the relatively simple task of verifying every individual in a squadron has completed mandatory security training. To complete this task, several elements are needed. The first of these is an accurate squadron roster, which in turn requires access to the AF personnel database via a regional server. Next a list of who has completed the training, which is kept up to date by the security manager via an Excel document stored on the shared drive. Access to the shared drive is also required. In this scenario all computers in the squadron access the network and thus the server via a single router. As can be seen in Figure 2.8, this example situation cannot be represented via a tree without duplicating the element that represents the router. Figure 2.9 shows how this duplication can be avoided if a true graph is used.

The difficulties surrounding the use of trees lead to a focus on various representations of a directed graph. This was reinforced by the fact that all other data structures used in computer science can be built or derived utilizing a directed graph structure [6]. Therefore, a directed graph of some form should be able to satisfy our needs. This research focuses on various representations of a directed graph: a radial view, a basic directed graph, and a force directed graph.

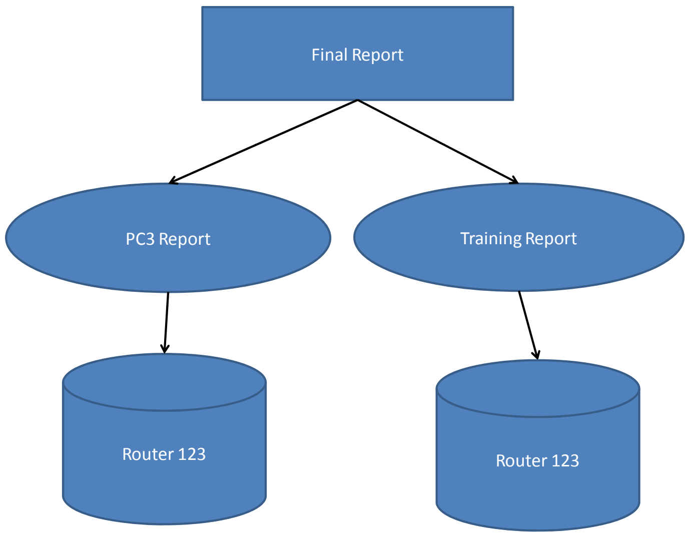

Figure 2.8: Four element example as a tree. Note that router 123 has to be repeated

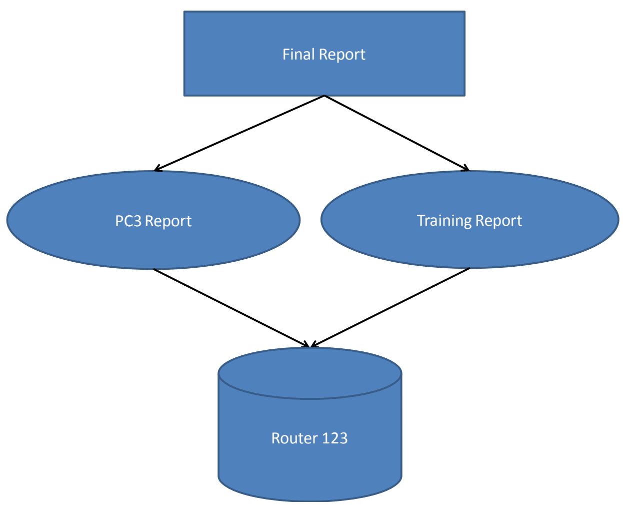

Figure 2.9: Four element example as a graph

2.5 Visualization Tools

There are a great number of Java graph visualization tool kits available. This research evaluated three to determine which was the most suitable for the needs of the research. In order to determine which three to evaluate, I looked for tools that claimed to include the following benefits. First, an auto layout capability in order to avoid writing a method to optimize placement of the visualization elements representing the graph manually. Second, the ability to handle large data sets in order to increase the scalability of the resulting visualization. Third, reported ease of use of the toolkit and documentation. As a result, this research looked at three toolkits. The first is a commercial product titled "JGraphpad Pro". The second toolkit to be examined is open source toolkit titled "JGraphT", which is optimized for large datasets. The third toolkit to be examined was `prefuse`, another open source toolkit.

2.5.1 JGraphpad Pro. JGraph was originally developed by Gaudenz Alder in 2000 and underwent further development via the open source community. JGraphpad Pro is a commercial product that according to the manufacturer includes all of the functionality of JGraph, tools for rapid development and deployment of visualization solutions and functionality for auto layout of the visualization [1]. These features avoid the necessity to write an algorithm to determine node placement or manually place all of the nodes and made this version of JGraph appear perfect for the research. However, I found the documentation to be poor and the toolkit difficult to use. As a result, I decided not to use this toolkit to produce the Visualization for this research.

2.5.2 JGraphT. JGraphT is an open source Java library designed to use JGraph for visualization. It expands JGraph to include additional mathematical Graph-Theory objects and algorithms and implements a simplified application programming interface (API). JGraphT is also optimized for data models and algorithms and designed to handle graphs with millions of vertices and edges. This library has the distinct advantage of being designed to handle very large data sets. However, as I learned during my evaluation of JGraphT, it uses the base JGraph library for its visualization. As such, the optimization and ability to handle millions of nodes is lost when the Graph is rendered into a visualization. This indicates the tool kit is very good where there is a large amount of data to parse and manipulate, but only a small subset of that data needs to be displayed in a visualization. Though this would meet the needs for the research, the functions the `prefuse` toolkit provided is better suited to the research.

*2.5.3 **`Prefuse`**: Information Visualization Toolkit.* The `prefuse` toolkit is available under the GNU license, meaning that it is copyrighted but it is free to use. It is designed to assist in the creation of interactive data visualizations. The toolkit includes sample code and auto layout functionality for a variety of visualizations including Trees, Tree maps, Directed Graphs, Flow Maps, as well as Radial and Fish-Eye visualizations. A paper presented at the *Conference On Human Factors in Computing Systems* in 2005 titled "`prefuse`: a toolkit for interactive information visualization" discribes the design as "The `prefuse` visualization framework" Figure 2.10 [9] and indicates the tool kit can handle at least 1.5 million nodes. Given the abundance of sample code, built in auto layout features and ability to process at least 1.5 million nodes, makes this tool kit is the most suitable of the three examined for this research.

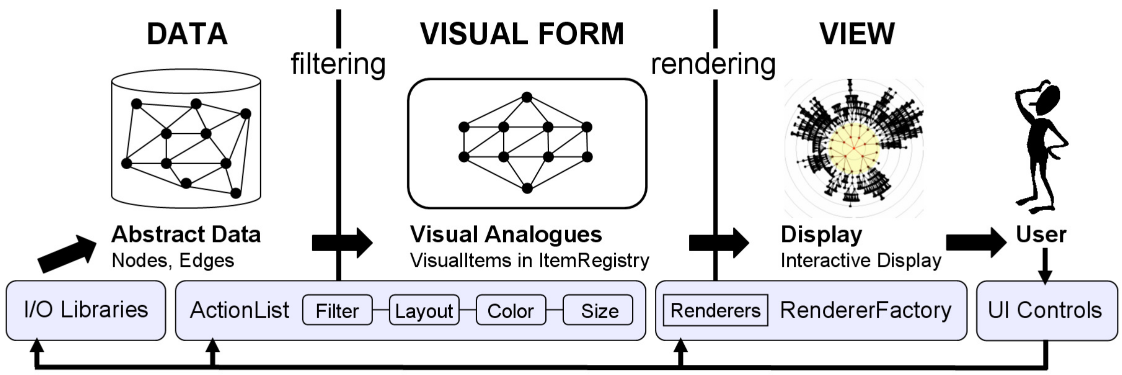

Figure 2.10: The `prefuse` visualization framework. Lists of composable actions filter abstract data into visualizable content and assign visual properties (position, color, size, font, etc). Renderer modules, provided on a per-item basis by a Renderer Factory, draw the Visual Items to construct interactive Displays. User interaction can then trigger changes at any point in the framework.

III. Methodology

This chapter covers the methodology for the research. It begins with an overview of the terminology that used to deal with the problem and this research's approach to it. Then a brief discussion concerning the source of the data used to create the visualizations and why this data was selected. The third section focuses on the visualization techniques chosen. The next section reviews the visualization toolkit selected. Finally, the last section consists of the explanation of the criteria chosen to evaluate the visualizations created.

Previous research on mission impact analysis visualization that included network equipment focused on physical representations, conceptual representations, or some hybrid of the two. Physical representations centered on the idea of creating a representation of the physical layout of the network. While conceptual representations selected some other organizational method other than physical location.

The creation of a system to display the physical layout and correlate network equipment to the equipment's physical location is based upon the belief that being able to show the physical relationship between network equipment makes it easier to understand the network. This belief has some merit since a router or switch shown to be in a building obviously provides connectivity for that building thus the building will be impacted by the failure of the equipment. However this is not always the case. Routers in a communications closet on the first floor may serve the first and third floors due to a low number of users on these floors, where an additional communications closet is needed on the second floor. Another reason to use the physical layout is the person looking at the visualization will be able to mentally overlay what they see onto mental images of the building, complex, base, etc. or a map of the area could be superimposed on the visualization itself.

Unfortunately, this solution brings with it a host of other problems that have not yet been adequately solved. Not the least of which is "How do you determine the placement of the equipment?" One solution is to annotate the location of each piece of equipment with building and room number, but this necessitates the manual placement of each visual representation and accurate maps to be digitized and used as a background. Another solution that is often used in physical layouts is to utilize Global Positioning System (GPS) coordinates and elevations. This system allows software to determine distances between each item, but it necessitates a new GPS reading every time a computer or piece of equipment is moved, or a GPS unit with wireless to be built into every piece of equipment and programmed to send in its location to some central server.

Even if all the overhead and added complexities of a physical layout are overcome, another problem is run into when attempting to include non-physical elements such as mission tasks or reports. Should the location of a report be listed as where it originates, where it is being sent, or the location of the data it is based on? Similar questions arise for mission tasks.

Conceptual representations overcome the difficulty of deciding where to place the physical location of non-physical elements, but have other difficulties. By abandoning physical location, some other organizational structure must be imposed. One of the possibilities for organizing the elements is based on who is responsible for the task or equipment. Another is who is using the equipment, needs the equipment operational, or task accomplished. Some researchers recommend stratifying the visualization into different categories with the physical network in one layer of the visualization, the systems in another, applications in third and so on [14,18]. Each of these requires more and more complex structures to store the data representing these elements with the various elements being broken up into more and more specialized logical compartments. These visualizations have the benefit that they display the information in a manner people are used to receiving the information. Anyone can understand the logic behind organizing elements based on work or data flow. In order to accomplish such a layout, complex data structures must be developed with many types of elements, for instance, one for routers and switches, another for links, a third for buildings, a forth for servers and so on. Or the elements must be manually placed by the operators.

This research begins by rejecting a fundamental assumption that both the physical layout and conceptual layout are based on. Specifically, the assumption that the visualization must be presented in such a manner as to allow a human to easily trace the interdependencies and determine how and when one element affects another. By rejecting this, the requirement to present a very complex spider web of interdependencies in a simplified manner disappears. With the elimination of this requirement the problems plaguing much of the research in this area also go away.

Instead, this research accepts that today's networks and the complex web of tasks that are dependent on the hardware have reached a point that is beyond most individuals to understand without careful tedious study no matter how it is formatted and displayed. From this view point the question shifts from "how do we display the dependencies in a meaningful visualization?" to "how can the cascade of failures and possible failures be determined and the result displayed in a meaningful visualization?" This is a question that is more in line with data manipulation and graph theory instead of node placement.

This method of looking at the problem allows this research to treat every element in the visualization the same, provided allowances are made for the fact that some elements may need to have different information associated with them. However, these differences are no longer a factor in the visualization itself. Instead every element must have three items of information in order to be displayed: a unique ID, a name, and a status indicator. Other information such as manufacturer, point of contact, and office name or symbol becomes optional. This in turn results in a solution that is extremely flexible and can be utilized no matter how policies and procedures change and will work even if the computer and network architecture undergoes a radical transformation resulting in equipment and dependencies that we cannot currently imagine.

Table 3.1: Example Actors

Physical	Router
Physical	Server
Physical	Data Link
Physical	Sensor
Physical	Computer
Physical	Electrical Generator
Physical	Building
Physical	Room
Physical	Report
$Non - Physical$	Missions
$Non - Physical$	Tasks
$Non - Physical$	Services
$Non - Physical$	Server Software
$Non - Physical$	Programs

3.1 Terminology

Actor: This term is used to identify any discrete element that is needed by another element or needs another element, and is represented by a node on the graph. Some examples of various types of actors are shown in Table 3.1, for clarity, both physical and non-physical actors are designated.

Need Line: Describes the relationship between two actors. If a mission requires a task to be completed for the mission's success, there would be a need line extending from the node representing the mission to the node representing the task, with the arrow pointing from the mission to the task, thus designating the mission's need of the task. For example, the mission "Provide E-mail Service" requires the "E-mail Server", among other things, to be functional. This relationship results in at least two actors (A: "Provide E-mail Service" and B: "E-mail Server") connected by a need line pointing from B to A.

3.2 Data

Eight data sets were utilized to test the viability of the visualization. Four were selected to test the visualization and program logic. Three are based on the scenario outlined in Chapter 1. The final data set is based on the mission requirements of an Air Mobility Element (AME).

The first three data sets include a set of actors whose dependencies form a loop, a set containing dual dependencies, and finally a set containing both dual dependencies and a loop. A fourth set containing a disconnected graph was also added. These test sets were selected to ensure that the visualization can handle the various levels of connectivity individually and together, then the visualization can handle a larger more complicated data set.

The next three sets were selected to demonstrate that the visualization solves the problems demonstrated by the situations in the scenario. The first of these is based on situation one. It concerns a UPS failing, shown in Figure 3.1. The second

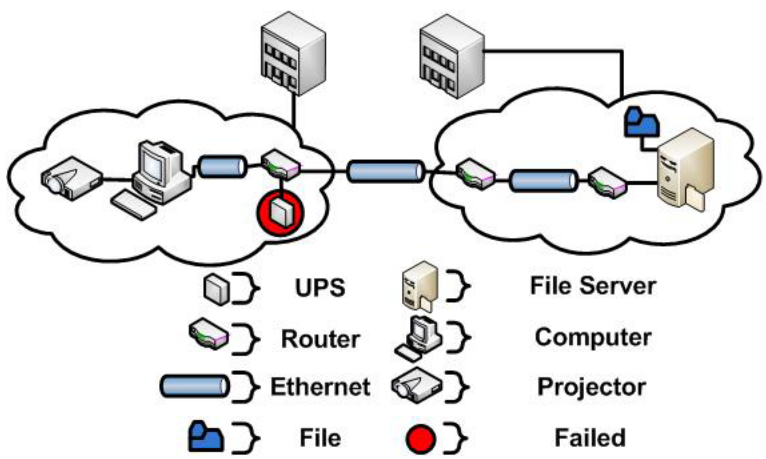

Figure 3.1: Illustrates the interconnectivity of the systems described in situation one of the scenario discussed in chapter 1.

concerns a router failing and is from situation two, shown in Figure 3.2. The third of the scenario data sets is based on the destruction of a building and its impact on base operations. To illustrate these, three generic missions were added. Each mission with network and/or phone requirements the resulting interdependency is shown in Figure 3.3. Specifically, that the visualization grants the NCC a quick and accurate information to as to the scope of the problem and its impact.

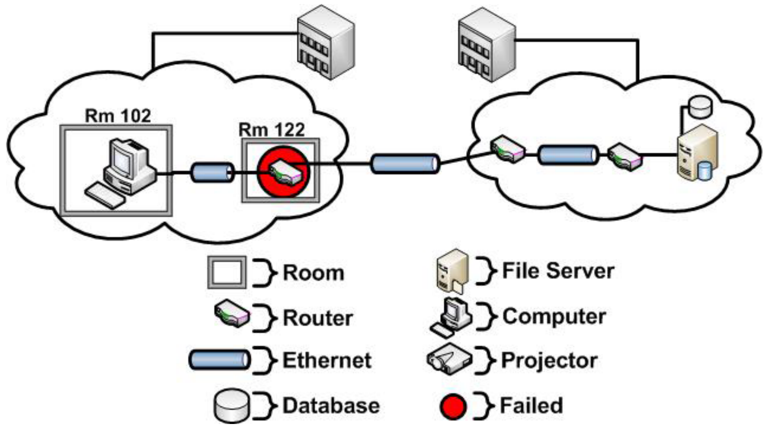

Figure 3.2: Illustrates the interconnectivity of the systems described in situation two of the scenario discussed in chapter 1.

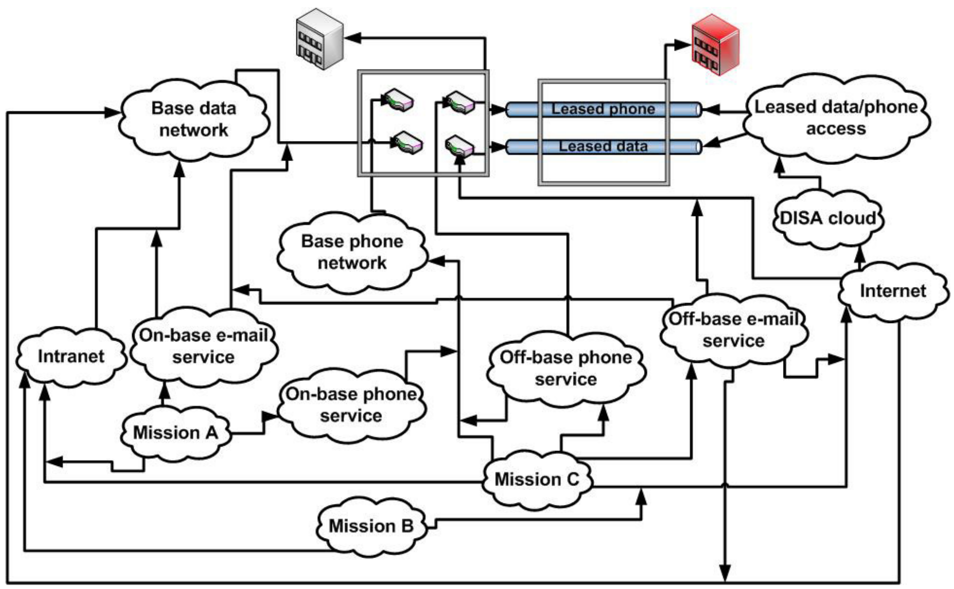

Figure 3.3: Illustrates the interconnectivity of a system based on the description of situation three of the scenario discussed in chapter 1.

The final set of test data was selected to demonstrate the visualization works with real world data. This data set is derived primarily from the research of Master Sargent Shaw in his thesis “A Model for Performing Mission Impact Analysis of Network Outages” [13]. Particularly, his “AME Multi-Layer Model”, shown in Figure 3.4, shows the interdependencies of the AME mission requirements.

This particular model abstracts away much of the infrastructure due to the sensitive nature of the network. In order to avoid unnecessarily increasing the classification of this thesis, a real infrastructure will not be utilized. A simplified infrastructure is substituted in order for visualizations to demonstrate the cascading failures that can occur with the loss of a router or other core piece of the network backbone. The details of each data set are discussed in Chapter 4.

3.3 Visualization Techniques

As discussed in Chapter 2, representing the data sets as trees results in added complications that are not necessary. With the elimination of tree based visualizations, the focus of this research is on visualizations based on directed graphs. More specifically, this research evaluates directed graphs, radial directed graphs, and force directed graphs along with minor variations to these, such as node and link-line coloration, node and link-line filtering, and interactive controls.

Furthermore, as discussed in the introduction of this chapter, instead of employing several different types of nodes in the graph, e.g. one for documents, one for hardware, a single node type is employed. The nodes represent all actors within the system and the links between the nodes represent the need lines. By reducing the nodes to this most fundamental level, the graphs become more complex, the added level of complexity can be visualized by imagining looking down through all the layers of the Multi-layer Network Centric Operations (NCO) Model, as shown in Figure 3.5, from Wong-Jiru's Thesis "Graph Theoretical Analysis of Network Centric Operations Using Multi-Layer Models" [18]. While this method results in a single graph composed of all of the nodes and links that are spread out over five graphs in Wong-Jiru's model, it does offer a benefit as well. Since the visualization system treats everything either as an actor or as a need line, the system can incorporate any future requirements. In other words, software based on this system would not need to be altered to account for a new way of doing things as technology progresses.

3.3.1 Evaluation Criteria. Each of the eight data sets will be used to test the various directed graph layouts. The resulting visualizations are evaluated qualitatively

Table 3.2: Criteria used for determine the 1-10 evaluation score for clarity, scope, and ease of use.

Evaluation Criteria	
Score	Criteria
1	Does not work at all
2	Meets very few needs, with a great deal of effort on the part of the user
3	Meets some needs, with a great deal of effort on the part of the user
4	Meets most needs, with a great deal of effort on the part of the user
5	Meets the minimum needs necessary to be usable
6	Meets needs with significant effort on the part of the user
7	Meets needs with moderate effort on the part of the user
8	Meets needs with some effort on the part of the user
9	Meets needs with very little effort on the part of the user
10	Works perfectly; accomplishing everything needed and desired

based on clarity, ease of use, and scope of information. Each graph is be assigned a score of 1-10 for each of these categories, with "1" being the lowest score and "10" the highest. These are qualitative measurements instead of quantitative because everyone will react to a visualization and color scheme slightly differently and I did not gather the opinions of a large enough group to have statistical significance. As a result, they are very subjective and represent a sliding scale with the lowest score representing "Does not work at all" and the highest "Works perfectly; accomplishing everything desired", as can be seen in Table 3.2.

"Clarity" refers to how clearly the visualization conveys the situational picture. For example, if a commander wishes to know if any individuals under his command are medically disqualified from deploying, a 2-5 page summery on the medical history of each individual would contain the requested information. However, it would not be very useful to the commander. Instead a simple list of names with a yes or no next to them with would be much clearer. This qualitative metric is used to evaluate how clearly the impact is conveyed in the various visualizations.

"Ease of use" represents how difficult it is for someone to utilize the visualization. This metric relates to how intuitive and easy to use the visualization is. An example of this type of comparison would be comparing Linux to Windows. While some many argue the technical merits of both operating systems, qualitatively many individuals find Windows easier to use than Linux [12]. Furthermore, most individuals feel the learning curve is much larger for Linux [12].

"Scope of information" is how much information is represented in the visualization. Returning to the example used for clarity, a list of names with a simple "yes" and "no" may accomplish the minimum scope needed to answer the commander's question. However, if you replaced the yes and no with yes, 3-6 weeks, and no, the commander would also know who will be cleared in 3-6 weeks even though they are not cleared currently. However, scope also takes into account how much information can be displayed in a manner easily dealt with. If instead of a single column of names you used two columns, the scope of a page would be almost doubled with no impact on usability. However, if you used six columns without dividers, even though there is six times as much information on the page it is now so cluttered as to not be beneficial. In this aspect scope affects clarity.

3.4 Toolkit

Three toolkits were evaluated, JGraphpad Pro, JGraphT and `prefuse`. I after evaluating them I rejected JGraphpad Pro and JGraphT. The primary reason I considered JGrpahpad Pro was the claim of the manufacturer that the tools offered rapid development and deployment of visualizations [1]. However, I found the documentation to be poor for JGraphpad Pro and the toolkit difficult to use. As a result I decided not to use this toolkit to produce the visualization for this research. JGrpahT was very promising and seemed to be the most scalable of the three toolkits I evaluated. However, as I learned during my evaluation of JGraphT, since it uses the base JGraph library for its visualization, the optimization and ability to handle millions of nodes is lost when the graph is rendered into a visualization. As a result this tool kit is very good where there is a large amount of data to parse and manipulate, but only the need to display a small subset of that data. Though this would meet the needs for the research, the toolkit `prefuse` provided is better suited to the research.

The `prefuse` toolkit was selected for several reasons. The primary reason was the automation that can be incorporated into the toolkit. The `prefuse` toolkit allows developers to create a basic visualization and quickly swap out data files with no changes to the base program that controls how that data will be displayed. A second reason was the auto placement algorithms it employs. Instead of coding the location of each node, the toolkit automatically attempts to place the nodes in a manner to make the resulting graph easier to view. Third, though the auto placement is not perfect, interactive controls can be placed in the visualization that allow the user to modify values utilized by the auto placement algorithms to adjust the placement of the nodes, and if this does not work the user can drag and manually place the nodes in most layouts. Fourth, the `prefuse` toolkit is well designed and open source. As a result there is a great deal of information about the product available on line.

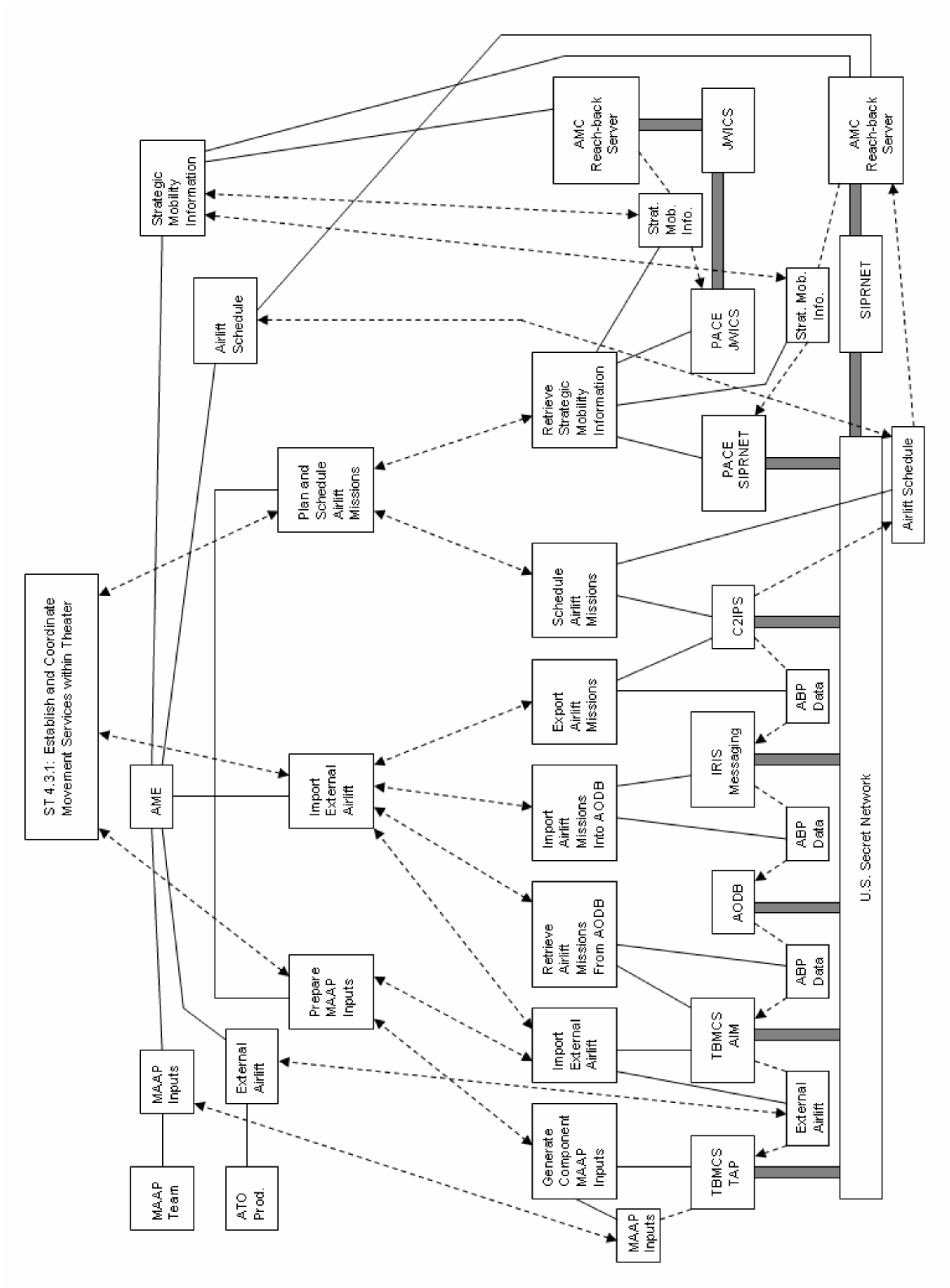

Figure 3.4: This model of the interdependencies of the elements necessary for the AME mission from the work of Master Sargent Shaw.

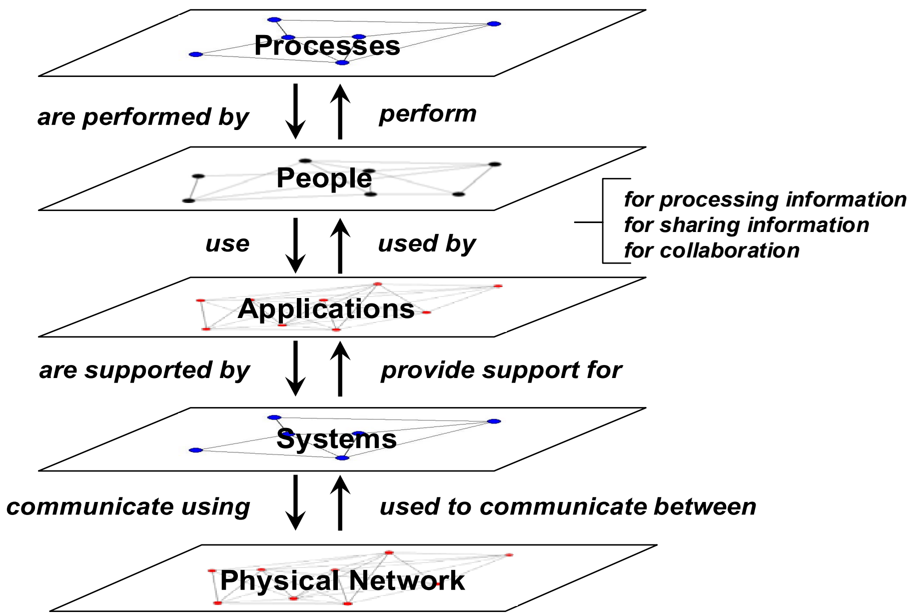

Figure 3.5: Multi-Layer NCO Model

IV. Comparative Analysis of Graph Visualizations

This chapter consists of analysis of the radial graph, directed graph and force directed graph visualizations using the eight datasets discussed in Chapter 3. Then this chapter continues on to review variations that can be applied to all three of these base graph visualizations to improve the overall visualization. These variations include highlighting direct dependents, actor details, and toggling of the actor's state. The sections for each dataset consist of the following information.

The sections are divided into three subsections discussing each of the three proposed graph types radial, directed, and force directed. The sections also contain an analysis of the graph based on its clarity, ease of use and scope. With the exception of the radial graph, the figures representing each of these graph views show the visualization as it initially loads the dataset with no direct manipulation unless otherwise stated.

The research analysis starts with a review of some of the fundamental graph conditions. I evaluate specific situations such as loops, dual dependencies, a dual dependent graph with looping, and a disconnected graph. Using these, I demonstrate the methodology used in this research can handle these special cases. I also examine the three situational scenarios discussed in Chapter 1 and finally a scenario based the actors and need lines of an actual USAF mission as identified in other research [14]. The primary point of the evaluation is to focus on utilizing the actor, need link structure to present a usable and effective visualization.

4.1 Dataset 1 - Loop

The loop dataset consists of five actors: A, B, C, D, and E. Actors A, B, and C have dependencies upon each other, such that it forms a loop. There is a need line running from actor A to show its dependency on actor B, actor B's dependency on actor C is shown by a need line, and actor C has a need line running to actor A. Actor B also has a need line to actor D which is outside the loop. Actor E is also outside the loop and is dependent on actor C. Actor D is also outside the loop and it depends on none of the other actors. This dataset is designed to demonstrate the visualization's ability to correctly handle dependency loops, and evaluate how they are visualized.

Figures 4.1, 4.2 and 4.3 show the first dataset as represented by the radial, directed, and force directed visualizations. In each of the three visualizations all actors are correctly colored based upon the dependency: the failed actor red, affected actors yellow, and unaffected actors green. The lines representing need lines correctly identify the direction of dependency. Since all three visualizations color each actor correctly it is known that the visualizations can successfully deal with datasets containing a loop. Given the size of this dataset it is unsurprising that the scope of the each visualization is complete and all of the data is displayed clearly.

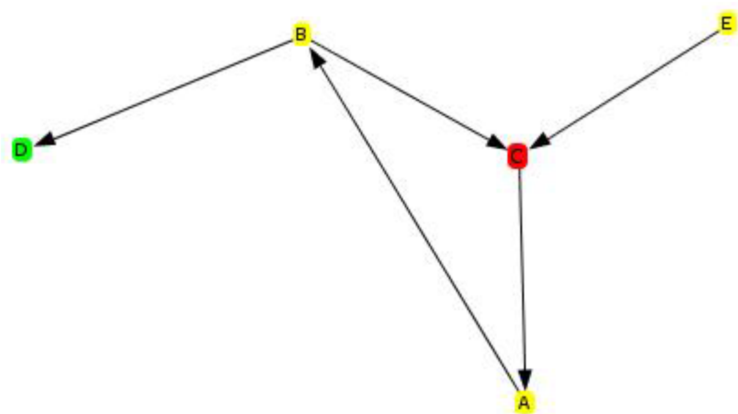

Figure 4.1: This diagram depicts the resultant radial visualization of the loop dataset. Actor C has been made the focus of the visualization because it represents the failed element.

4.1.1 Radial Graph: Dataset 1 - Loop. As shown in Figure 4.1 with the failed actor selected as the focus, all of the need lines display properly and nothing is obscured. The radial visualization of the loop dataset proves to be very easy to use. Clicking on the actor you wish to have focus is intuitive. Actors can be dragged to other positions, though this positioning is no longer relevant once a new focus is selected. The qualitative metric for the radial visualization of the loop dataset is shown in Table 4.1

Table 4.1: Clarity, scope, and ease of use metrics for the visualizations of Dataset 1

Dataset 1			
	Radial	Directed	Force Directed
Clarity	9	9	9
Scope	9	9	9
Ease of use	8	8	7

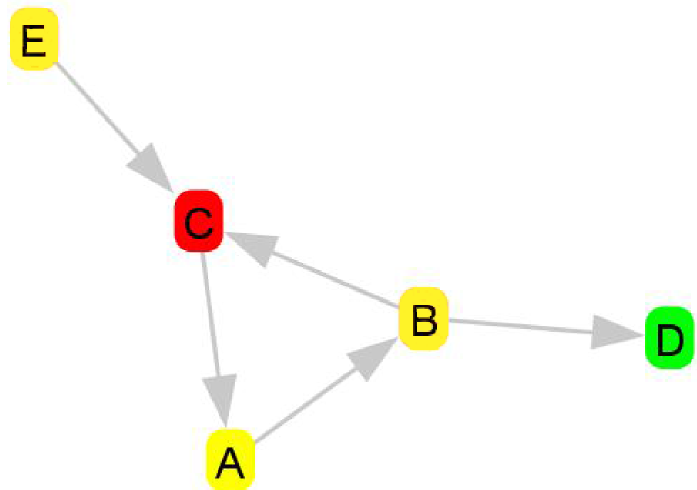

Figure 4.2: This diagram depicts the resultant directed visualization of the loop dataset. Actor C represents the failed element.

4.1.2 Directed Graph: Dataset 1 - Loop. As shown in Figure 4.2 the directed visualization of the loop dataset presents a very clear picture of the situation. This visualization also proved very easy to use, actors can be dragged to new positions. The lack of focus redrawing results in actors staying where placed for later reference, which is an improvement over the radial visualization, but the loss of the ability to focus on an actor offsets the benefit. The qualitative metric for the radial visualization of the loop dataset is shown in Table 4.1

4.1.3 Force Directed Graph: Dataset 1 - Loop. The force directed visualization yields a very similar depiction of dataset one the directed visualization as can be seen in Figure 4.3. The default force settings results in the actors being placed so as to avoid any overlapping, but the controls do add additional complication to

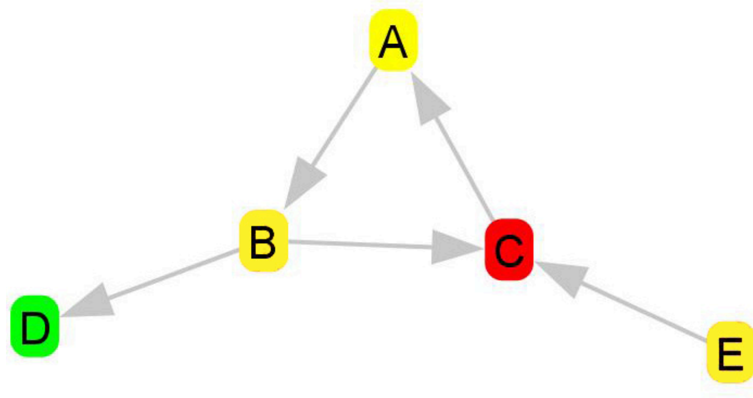

Figure 4.3: This diagram depicts the resultant force directed visualization of dataset 1. Actor C represents the failed element.

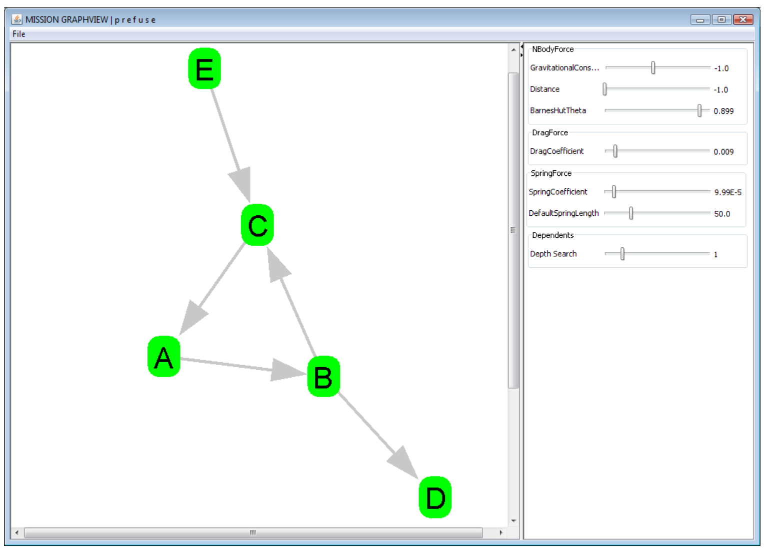

Figure 4.4: This diagram depicts the resultant force directed visualization of dataset 1, as well as the force controls for the visualization.

the interface as can be seen in Figure 4.4. As Figure 4.5 shows, the force control panel is made up of three sections. The first is the nbody control, this controls how the the gravity acts on the nodes representing actors. The “gravitationalconstant” control determines, as one might expect, the gravitational constant for the visualization. “Distance” is also relatively clear, determining the distance over which the gravitational field of each actor extends. The “barnshuttheta” control is not as self explanatory. `Prefuse` utilizes the Barnes-Hut algorithm in its force simulator. This algorithm was originally developed to assist in modeling things such as planetary and stellar systems. [3] It is used here to allow the user to fine tune the auto-layout, as are all of the other force controls. The "dragforce" control modify the simulated friction of the surface of the visualization on the actors. The "springforce" controls the characteristics of the need lines, effecting the length of the need line and how much the pull or push against actors. This seems complicated but the user can adjust these via trial and error, and since the visualization is interactive in near real time it does not take long to discover a setting that works for a given dataset. This has resulted in the visualization's score in the ease of use metric being poorer than the other visualizations but not significantly so, as shown in Table 4.1.

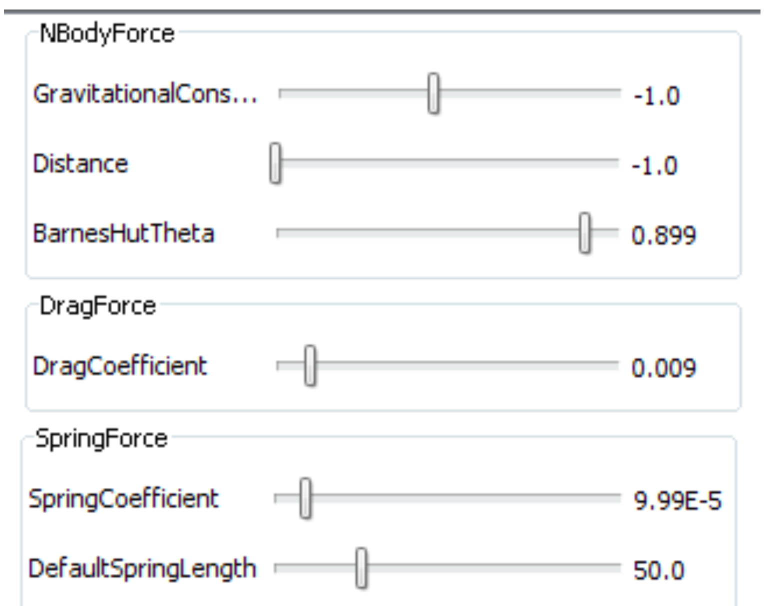

Figure 4.5: This depicts a close up of the force controls for the force directed visualization. The Nbody force section controls how the the gravity acts on the nodes representing actors. The Drag force is a determines the effect of drag on the individual elements and the graph as a whole. The spring force controls effect the lines/arrows representing need lines.

4.2 Dataset 2: Dual Dependency

This dataset consists of five actors: A, B, C, D, and E. Actor A is dependent on B, B is dependent on C. C is dependent on both B and D. D is dependent on C and

E. E is not dependent on any of the other actors. This dataset is designed to demonstrate the visualization's ability to correctly handle graphs with dual dependencies and evaluate how they are visualized.

Figures 4.6, 4.7, and 4.8 show dataset two as represented by the radial, directed, and force directed visualizations. As can be seen, just as with dataset one, in each of the three visualizations all actors are correctly colored based upon the dependency: the failed actor red, affected actors yellow, and unaffected actors green. The arrows representing need lines correctly identify the direction of dependency. The fact that they all correctly color the actors demonstrates the visualization is capable of handling datasets that contain dual dependences. Given the size of this dataset it is unsurprising that the scope of the each visualization is complete.

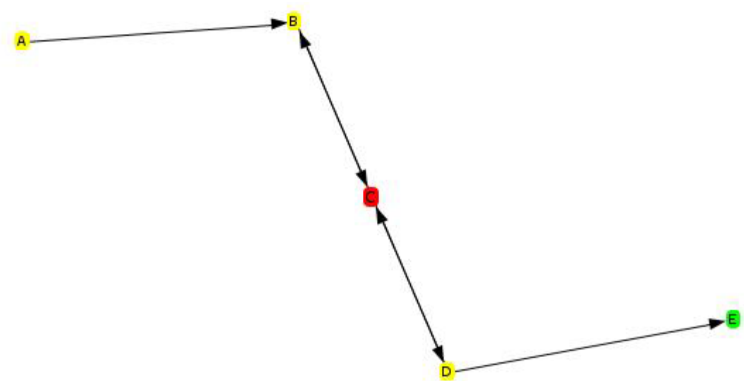

Figure 4.6: This diagram depicts the resultant radial visualization of dataset 1. Actor C has been made the focus of the visualization because it represents the failed element.

4.2.1 Radial Graph: Dataset 2 - Dual Dependency. Figure 4.6 shows that with the failed actor is selected as the focus, all of the need lines display properly and nothing is obscured. The radial visualization of the dataset proves to be very easy to use. Clicking on the actor you wish to have focus is intuitive. Actors can be dragged to other positions, though this positioning is no longer in effect once a new focus is selected. The qualitative metric for the radial visualization of the dual dependent dataset is shown in Table 4.2

Table 4.2: Clarity, scope, and ease of use metrics for the visualizations of Dataset 2

Dataset 1			
	Radial	Directed	Force Directed
Clarity	9	9	9
Scope	9	9	9
Ease of use	8	8	7

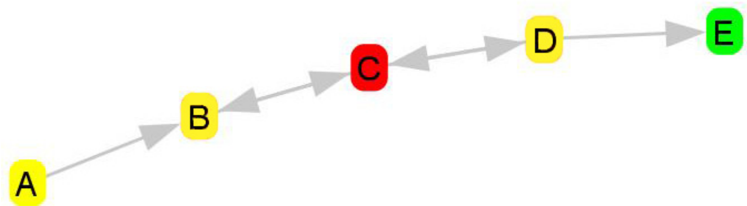

Figure 4.7: This diagram depicts the resultant directed visualization of dataset 2. Actor C represents the failed element.

4.2.2 Directed Graph: Dataset 2 - Dual Dependency. Figure 4.7 shows the directed visualization of the actors with dual dependencies in the dataset presents a very clear picture of the situation. This visualization also proved very easy to use, actors can be dragged to new positions. The lack of focus redrawing, results in actors staying where placed for later reference, which is an improvement over the radial visualization but the loss of the ability to focus on an actor balances the benefit. The qualitative metric for the radial visualization of the dual dependent dataset is shown in Table 4.2

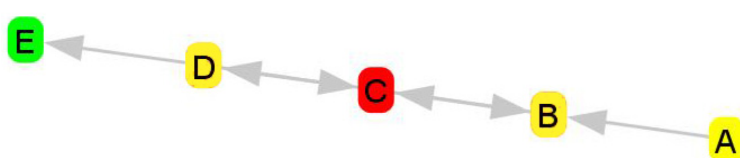

Figure 4.8: This diagram depicts the resultant force directed visualization of dataset 2. Actor C represents the failed element.

4.2.3 Force Directed Graph: Dataset 2 - Dual Dependency. As can bee seen in Figure 4.8, the force directed visualization yields a very similar depiction of dataset one the directed visualization. The default force settings results in the actors being placed so as to avoid any overlapping, but the controls do an additional complication to the interface. This has resulted in the visualization's score in the ease of use metric being poorer than the other visualizations, as shown in Table 4.2.

4.3 Dataset 3: Loop with Dual Dependency

The loop with dual dependency dataset also consists of five actors: A, B, C, D, and E. Actors A, B, and C form a loop with actor A dependent on actor B, actor B dependent on actor C and actor C dependent on actor A. Actor B is dependent on actor A and is also dependent on actor D which is outside the loop. Actor E is also outside the loop and dependent on actor C. Actor D is also outside the loop and it depends on none of the other actors. This dataset is designed to demonstrate the visualizations ability to correctly handle dependency loops containing a dual dependency, and evaluate how they are visualized.

Figures 4.9, 4.10 and 4.11 show dataset three as represented by the radial, directed, and force directed visualizations. In each of the three visualizations all actors are correctly colored based upon the dependency: the failed actor red, affected actors yellow, and unaffected actors green. The lines representing need lines correctly identify the direction of dependency. This demonstrates the visualization can correctly cope with a dataset with a loop containing a dual dependency. Given the size of this dataset it is unsurprising that the scope of the each visualization is complete.

4.3.1 Radial Graph: Dataset 3 - Loop with Dual Dependency. Figure 4.9 shows that with the failed actor is selected as the focus, all of the need lines display properly and nothing is obscured. The radial visualization of the dataset containing a loop with a dual dependency proves to be very easy to use. Clicking on the actor you wish to have focus is intuitive. Actors can be dragged to other positions, though this

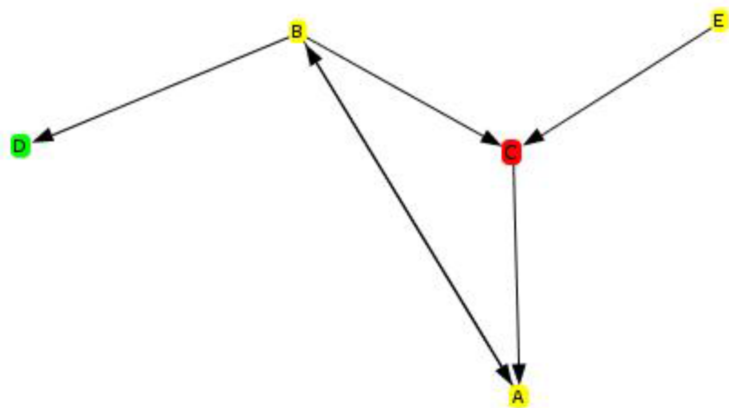

Figure 4.9: This diagram depicts the resultant radial visualization of dataset 3. Actor C has been made the focus of the visualization because it represents the failed element.

Table 4.3: Clarity, scope, and ease of use metrics for the visualizations of Dataset 3

Dataset 3			
	Radial	Directed	Force Directed
Clarity	9	9	9
Scope	9	9	9
Ease of use	8	8	7

positioning is no longer in effect once a new focus is selected. The qualitative metric for the radial visualization of the dual dependent dataset is shown in Table 4.3.

4.3.2 Directed Graph: Dataset 3 - Loop with Dual Dependency. The directed visualization of dataset three, shown in Figure 4.10, confirms that the directed graph is easy to use. The reasoning and influencing factors for the scores shown in Table 4.3, are identical to those that resulted the scores shown in Table 4.2.

4.3.3 Force Directed Graph: Dataset 3 - Loop with Dual Dependency. The force directed visualization of dataset three, shown in Figure 4.10, did not require any modification of the default force settings. Due to the simplicity of this dataset from a visual perspective, the reasoning and influencing factors for the scores shown in Table 4.3, are identical to those that resulted the scores shown in Table 4.2.

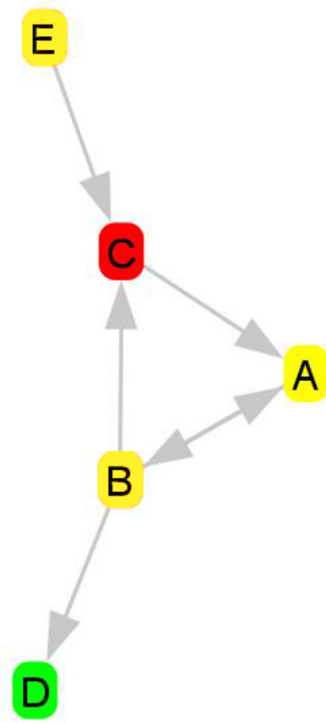

Figure 4.10: This diagram depicts the resultant directed visualization of dataset 3. Actor C represents the failed actor.

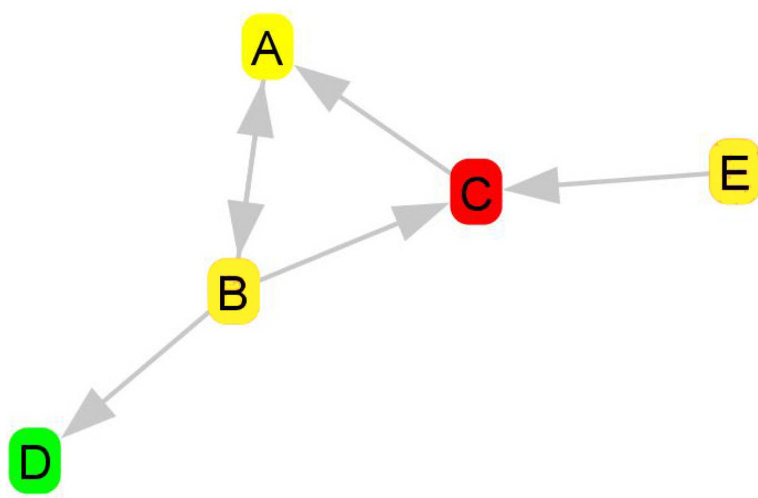

Figure 4.11: This diagram depicts the resultant force directed visualization of dataset 3. Actor C represents the failed element.

4.4 Dataset 4: Disconnected

The disconnected dataset consists of ten actors forming 2 disconnected sub-graphs. The first sub-graph consists of actors A, B, C, D, and E. Actors A, B, and

C form a loop with actor A dependent on actor B, actor B dependent on actor C and actor C dependent on actor A. Actor B is also dependent on actor D which is outside the loop. Actor E is outside the loop and is dependent on actor C. Actor D is also outside the loop and is dependent on none of the other actors. The second sub-graph consists of actors F, G, H, I, and J. Actors F, G, and H form a loop with actor F dependent on actor G, actor G dependent on actor H and actor H dependent on actor F. Actor G is also dependent on actor I which is outside the loop. Actor J is also outside the loop and dependent on actor H. Actor I is also outside the loop and depends on none of the other actors. This dataset is designed to demonstrate the visualization's ability to correctly handle disconnected graphs, and evaluate how they are visualized.

Figures 4.13, 4.15, and 4.16 show dataset three as represented by the radial, directed, and force directed visualizations. In each of the three visualizations all actors are correctly colored based upon the dependency: the failed actors red, affected actors yellow, and unaffected actors green. The need lines correctly identify the direction of dependency. This demonstrates the visualization can correctly cope with a dataset that results in a graph that is made of two or more non-connected subgraphs. Given the size of this dataset it is unsurprising that the scope of the each visualization is complete.

4.4.1 Radial Graph: Dataset 4 - Disconnected. The radial visualization of the disconnected dataset presents a poor picture of the situation. When the visualization first positions the actors it placed all of the actors of one of the subgraphs at the same location (stacked) as can be seen in Figure 4.12. Once the stack of actors is dragged apart and the J actor is given the focus, the J actor is brought to the center of the display. Then when the C actor is given the focus, the display overlaps the C actor directly over the J actor resulting in a visualization that appears incorrect, shown in Figure 4.13. This stacking of actors changes the visual meaning of this display. The misleading overlapping, though correctable by dragging the actors apart

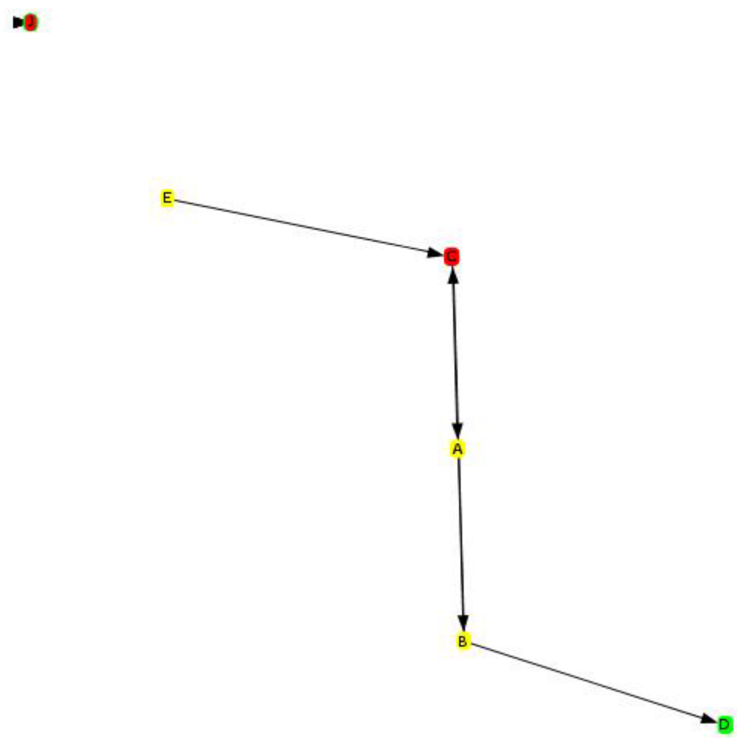

Figure 4.12: This diagram depicts the resultant radial visualization of dataset 4 in a radial display. Actor J was selected as the focus, then actor C was selected as both represent failed actors. This display has moved actor C over actor J, resulting in a misleading visualization.

as is shown in Figure 4.14, as well as the need to separate the actors adversely effects the visualization's scores for this dataset as can be seen in Table 4.4.

4.4.2 Directed Graph: Dataset 4 - Disconnected. The directed visualization of the fourth dataset also presents a poor picture of the situation. The visualization has difficulty with the disconnected nature of the graph and as can be seen by Figure 4.13, the graphs default to positions that make it difficult to identify that they are not connected. Even though this could be easily corrected by dragging the actors to new locations, it adversely effected the directed visualization in relation to dataset four reflected in Table 4.4.

4.4.3 Force Directed Graph: Dataset 4 - Disconnected. Unlike the radial and directed visualizations the force directed visualization displayed dataset four with the

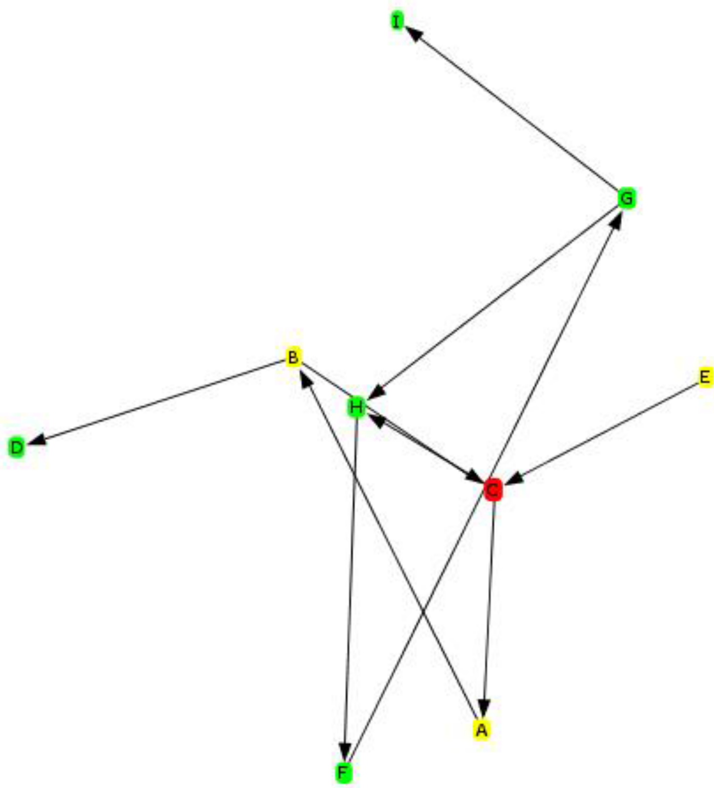

Figure 4.13: This diagram depicts the resultant radial visualization of dataset 4. Actor J was selected as the focus, then actor C was selected as both represent a failed element.

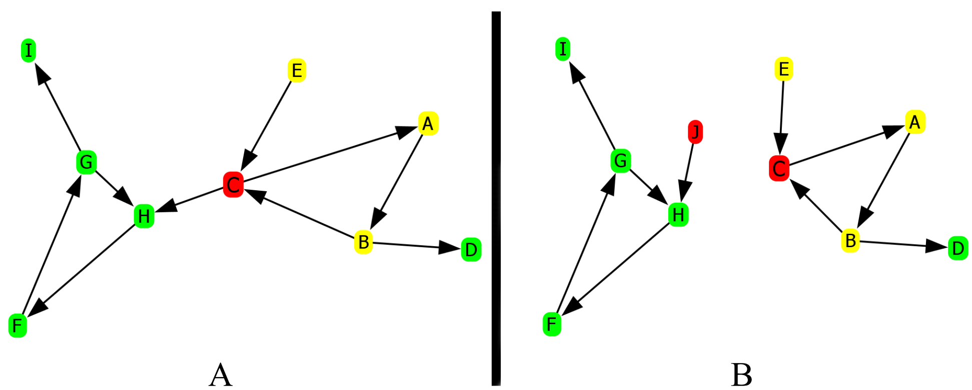

Figure 4.14: This diagram depicts the radial visualization of dataset 4. A) Shows the visualization part way through separating the disconnected graphs. B) shows the final separation.

same clarity it displayed the others. As can be seen in Figure 4.16, the two subgraphs automatically separated. Some minor adjustments to the force settings were

Table 4.4: Clarity, scope, and ease of use metrics for the visualizations of Dataset 4

Dataset 4			
	Radial	Directed	Force Directed
Clarity	6	7	9
Scope	8	8	9
Ease of use	8	8	7

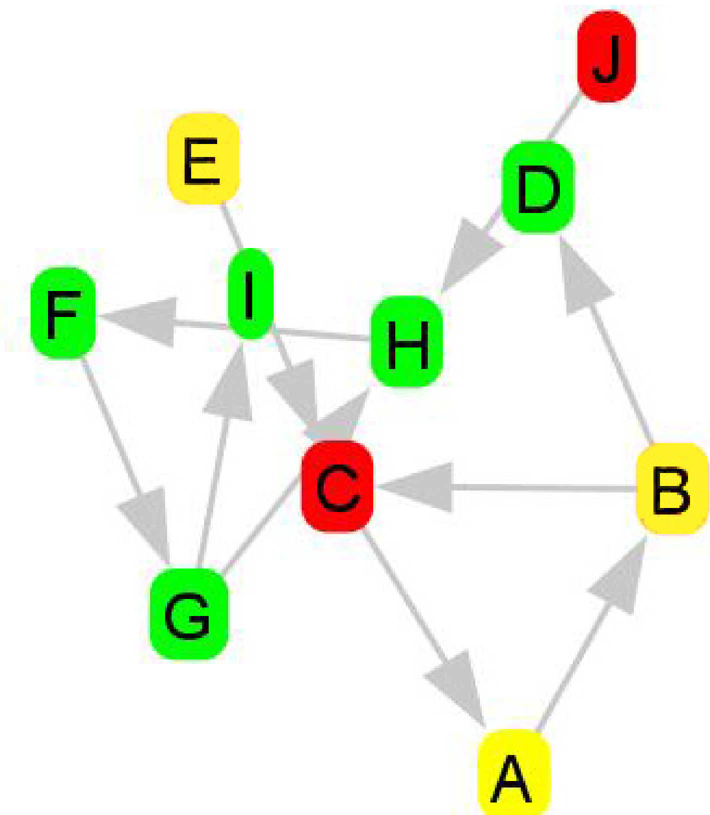

Figure 4.15: This diagram depicts the resultant directed visualization of the disconnected dataset. Actor J and C represent the failed elements.

needed to keep the sub-graphs near each other but this was easily accomplished. This consistency is reflected in the visualization's scores, as shown in Table 4.4.

4.5 Dataset 5: Situation 1 "The Theater"

This dataset consists of twelve actors and represents the scenario's first situation described in Chapter 1. This dataset is designed to demonstrate the visualization's

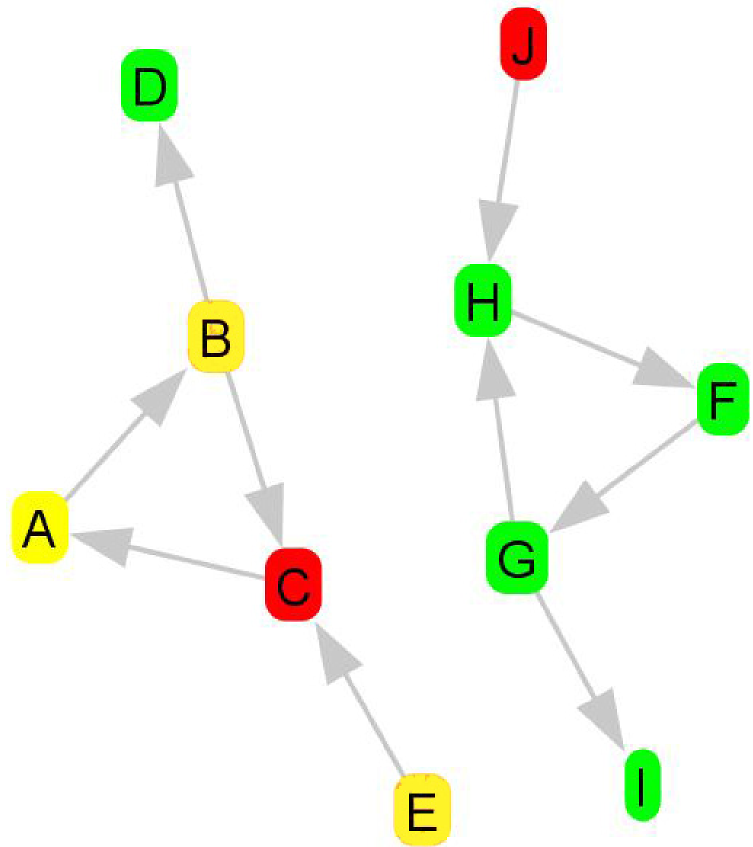

Figure 4.16: This diagram depicts the resultant force directed visualization of dataset 4. Actors J and C represent the failed elements.

ability to model this situation using the radial, directed, and force directed visualizations and determine their effectiveness.

Figures 4.17, 4.18, and 4.19 show dataset five as represented by the radial, directed, and force directed visualizations. As with previous datasets the actors and need lines are displayed correctly.

Dataset five consists of 12 actors, shown in Table 4.5, and their dependencies. The mission is dependent on the projector, computer, file, and theater link. The theater link is dependent on the theater router. The theater router is dependent on the base link, and the theater router's UPS. The communications building router is dependent on the base link. The Communications building link is dependent on the

Table 4.5: Dataset 5 contains the 12 actors shown here

Dataset 5: actors	
1. Theater Mission	7. Projector
2. Computer	8. File "The Star-Spangled Banner"
3. Theater link "The Ethernet link in the theater"	9. Theater router
4. Theater router UPS	10. Base link
5. Communications building router	11. Communications building link
6. Server router	12. File server

communications building router. The server router is dependent on the communications building link. The file server is dependent on the server router.

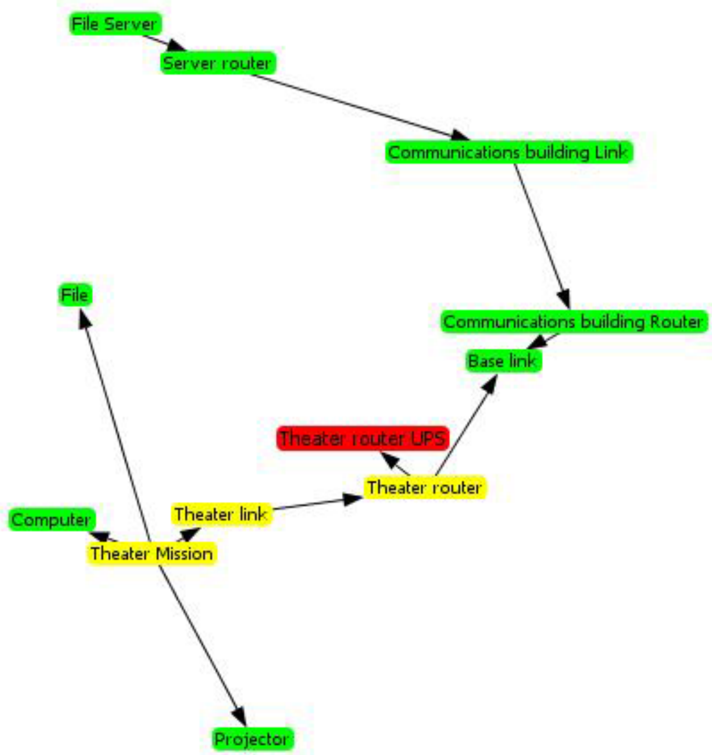

Figure 4.17: This diagram depicts the resultant radial visualization of dataset 5. The actor designated "Theater router UPS" was selected as the focus as it represents the failed element.

4.5.1 Radial Graph: Dataset 5 - "The Theater". The radial visualization of the theater's dataset presents a clear picture of the situation. Prior to selecting the actor designated "Theater router UPS" some of the need lines were obscured but, once this actor was selected, was no longer the case, as is shown in Figure 4.17. This

Table 4.6: Clarity, scope, and ease of use metrics for the visualizations of Dataset 5

Dataset 5			
	Radial	Directed	Force Directed
Clarity	8	7	9
Scope	8	8	9
Ease of use	8	8	7

resulted in a higher score for clarity than in dataset four, but a lower score than for datasets 1-3 shown in Table 4.6.

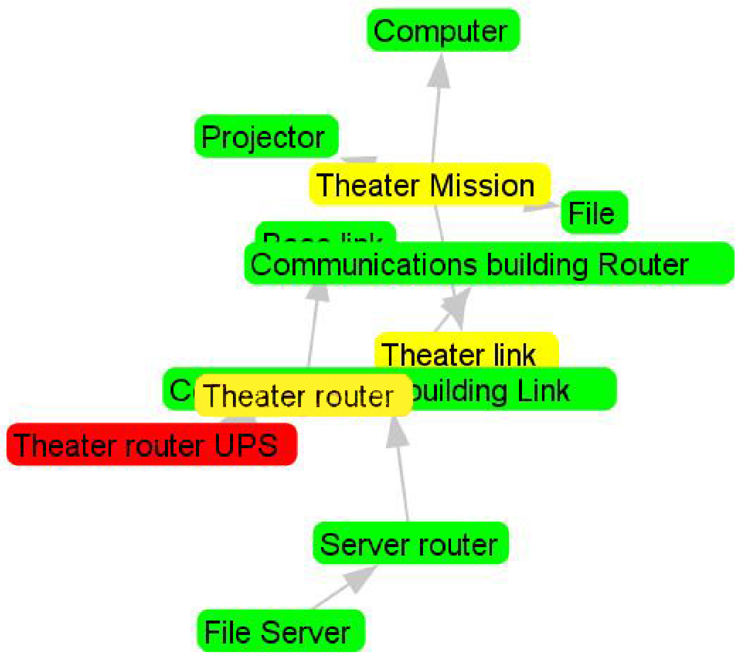

Figure 4.18: This diagram depicts the resultant directed visualization of the theater dataset. The actor designated "Theater router UPS" represents the failed element.

4.5.2 Directed Graph: Dataset 5 - "The Theater". The directed visualization of the fifth dataset, Figure 4.18, also presents a good picture of the situation. However, visual cluttering of the graph is beginning to imply that for large, interconnected graphs, a great deal of manual adjusting of the actors is needed as is shown by the scores in Table 4.6. However, this does not yet detract from the visualization.

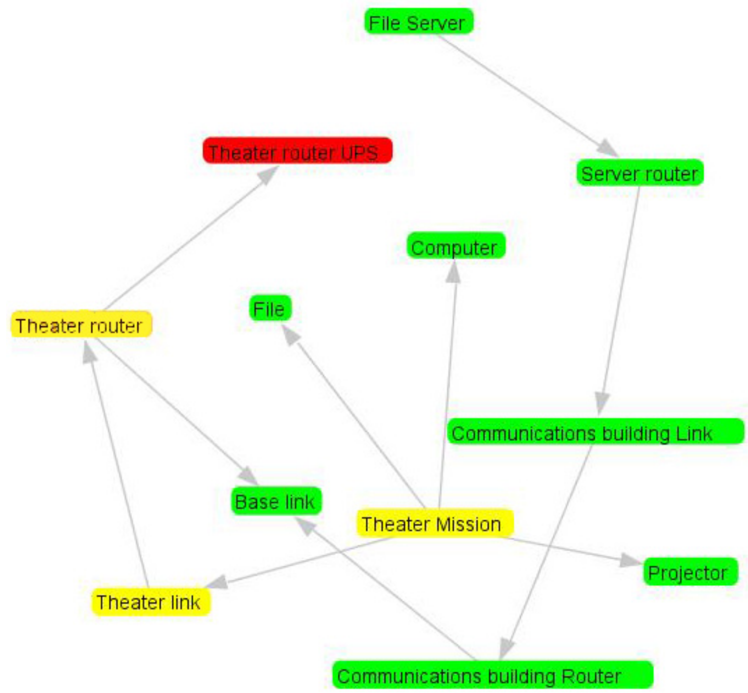

Figure 4.19: This diagram depicts the resultant force directed visualization of dataset 5. The actor designated "Theater router UPS" represents the failed element.

4.5.3 Force Directed Graph: Dataset 5 - "The Theater". The force directed visualization of the theater dataset presents a good picture of the situation as well. Though the force directed visualization, shown in Figure 4.19, did not place the actors in the optimum position to avoid overlap, the resulting graph is clear and easy to understand and was not detrimental to the development of a good grasp of the situation represented by the visualization. This fact resulted in the scores for the force directed visualization to remain consistent shown in Table 4.6.

4.6 DataSet 6: Situation 2 - "PMI"

This dataset consists of eleven actors, as is shown in Table 4.7, and represents the scenario's second situation, described in Chapter 1. The dataset is designed to demonstrate the visualization's ability to model the situation using the various graphs.

Table 4.7: Dataset 6 contains the 11 actors shown here

Dataset 6: actors	
1. PMI Mission	7. Database client software
2. Computer	8. Building 25 link
3. Building 25 router	9. Base link
4. Building 375 router	10. Building 375 link
5. Server router	11. Database server
6. Tool database	

The need lines represent the dependencies between the actors. The PMI mission is dependent on the database client software, computer, and the building 25 link. The database client software is dependent on the building 25 link, the computer, the tool database, and the database server. The computer is dependent on the building 25 link. The building 25 link is dependent on the building 25 router. The building 25 router is dependent on the base link. The building 375 router is dependent on the base link. The building 375 link is dependent on the building 375 router. The server router is dependent on the building 375 router. The database server is dependent on the server router. The tool database is dependent on the database server.

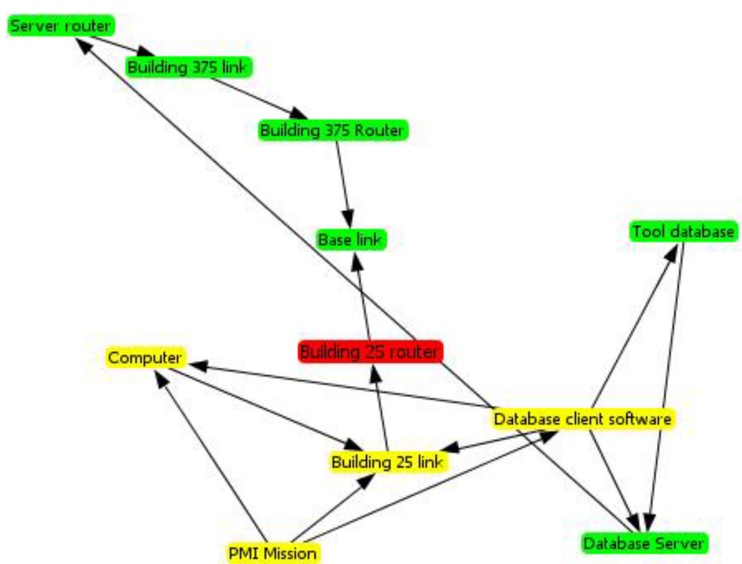

Figure 4.20: This diagram depicts the resultant radial visualization of dataset 6. The actor designated "Building 25 router" was selected as the focus as it represents the failed element.

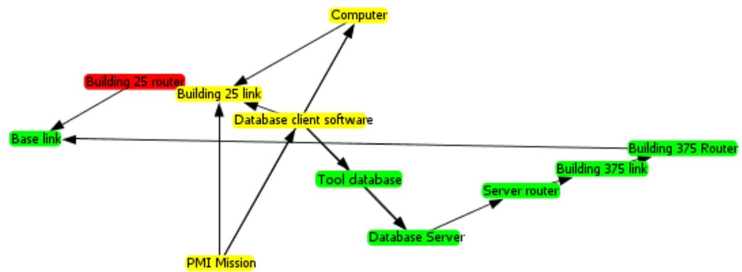

Figure 4.21: This diagram depicts the resultant radial visualization of data set 6. Utilizing a radial visualization. The need line linking the PMI mission and computer is obscured.

Table 4.8: Clarity, scope, and ease of use metrics for the visualizations of Dataset 6

Dataset 6			
	Radial	Directed	Force Directed
Clarity	8	7	9
Scope	8	8	9
Ease of use	8	8	7

4.6.1 Radial Graph: Dataset 6 - "PMI". The radial visualization for the sixth dataset shows indication that radial visualizations for highly interdependent datasets will result in a view slightly better than a normal directed graph, this effect can be seen by comparing Figures 4.20 and 4.22. In this situation things are relatively clear when the actor designated "Building 25 router" is used as a focus, but many of the other actors when selected as focus result in a visualization with overlaps as is shown in Figure 4.21 where the need line representing the PMI mission's need of the computer is obscured. These factors result in the scores listed in Table 4.8.

4.6.2 Directed Graph: Dataset 6 - "PMI". The directed visualization for the PMI dataset indicates that directed visualizations for highly interdependent datasets will result in a view that requires extensive manual adjustment, as can be seen in Figure 4.22. This confirms the observations concerning the cluttering of a directed visualization as interdependences within the dataset increases, based on Figure 4.18.

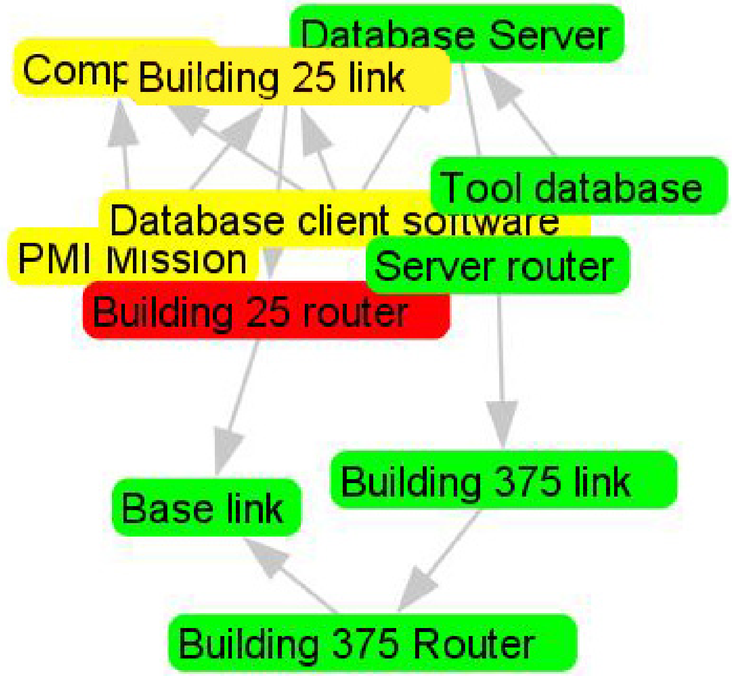

Figure 4.22: This diagram depicts the resultant directed visualization of dataset 6. The actor designated "Building 25 router" represents the failed element.

The level of clustering is not enough to justify decreasing the visualization's score below where it was for dataset five, this results in the scores listed in Table 4.8.

4.6.3 Force Directed Graph: Dataset 6 - "PMI". The force directed visualization of dataset six creates a good situational picture, shown in Figure 4.23. The force settings needed to be adjusted to avoid cluttering; the resulting visualization is very clear and easy to understand. The clarity, ease of use and scope of the forced visualization for dataset six and the other datasets are nearly identical resulting in the same score as the others, shown in Table 4.8.

4.7 Dataset 7: Situation 2 - "Building 74 Destruction"

The dataset representing the destruction of building 74 consists of twenty-three actors, shown in Table 4.9, and represents the third scenario situation described in Chapter 1. The dataset is designed to demonstrate the visualization's ability to

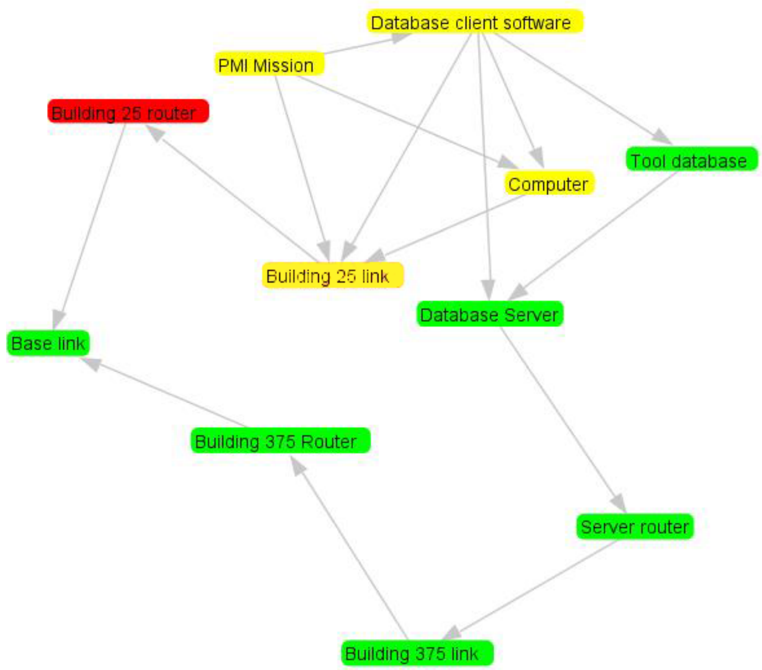

Figure 4.23: This diagram depicts the resultant force directed visualization of dataset 6. The actor designated "Building 25 router" represents the failed element.

model this situation using the various graphs, and demonstrate its ability to handle a large and more complex dataset. There are thirty-nine direct dependencies for these elements, derived from Figure 3.3. For a detailed list see Appendix A.

4.7.1 Radial Graph: Dataset 7 - "Building 74 Destruction". The radial visualization for the seventh dataset demonstrates that radial visualization for highly interdependent datasets results in a view that is cluttered and distracting. In this situation there is no overlapping of actors when the actor designated "Building 74" is used as a focus, shown in Figure 4.24. However, the overlapping of need lines between the actors results in a visualization that is significantly less clear than the radial visualization of dataset 6 in Figure 4.20. As a result, the visualization's score, shown in Table 4.10, suffers in the area of clarity.

4.7.2 Directed Graph: Dataset 7 - "Building 74 Destruction".

Table 4.9: Dataset 7 contains the 23 actors shown here

Dataset 7: actors	
1. Mission A	13. Mission B
2. Mission C	14. Intranet
3. Internet	15. On-base e-mail service
4. On-base phone service	16. Off-base e-mail service
5. Off-base phone service	17. Base data network
6. Base phone network	18. DISA cloud
7. Leased data/phone access	19. Building 5
8. Building 5 Rm 232	20. Building 74
9. Building 74 Rm 1	21. Leased data line
10. Leased phone line	22. Outbound phone equipment
11. Outbound data equipment	23. Internal phone equipment
12. Internal data equipment	

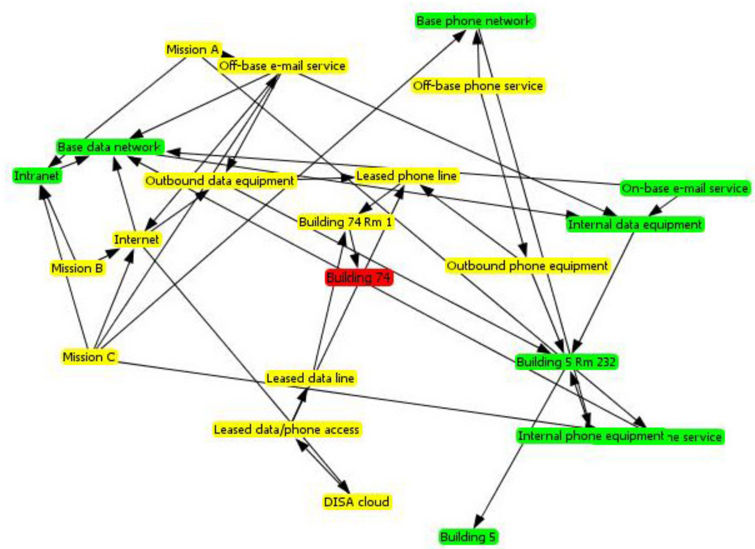

Figure 4.24: This diagram depicts the resultant radial visualization of dataset 7. The actor designated "Building 74" was selected as the focus as it represents the failed element.

4.7.2.1 Clarity. The directed visualization for the seventh dataset confirms that directed visualizations for highly interdependent datasets results in a view that requires extensive manual adjustment (Figure 4.25). Without this adjustment, the visualization does not provide a clear picture of the situation. It also indicates that clarity and scope of the visualization continues to degrade as interdependencies increase. These facts result in the ratings shown in Table 4.10.

Table 4.10: Clarity, scope, and ease of use metrics for the visualizations of Dataset 7

Dataset 7			
	Radial	Directed	Force Directed
Clarity	7	7	9
Scope	8	7	9
Ease of use	8	8	7

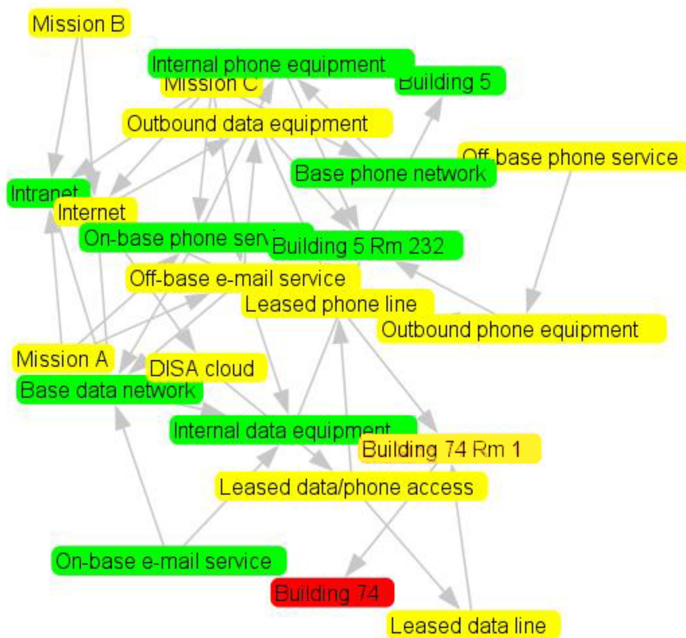

Figure 4.25: This diagram depicts the resultant directed visualization of dataset 7. The actor designated "Building 74" represents the failed element.

4.7.3 Force Directed Graph: Dataset 7 - "Building 74 Destruction". The force directed visualization of the destruction of building 74 dataset, shown in Figure 4.26, suffers the same deficiencies and the same strengths as have been observed with every dataset displayed using the forced directed visualization. This trend dictates that the force directed graph may be the best of the three for general use, though in specific situations directed or radial may be better. The scores for the clarity, scope, and ease of use of the force directed visualization in relation to dataset 7 are shown in Table 4.10

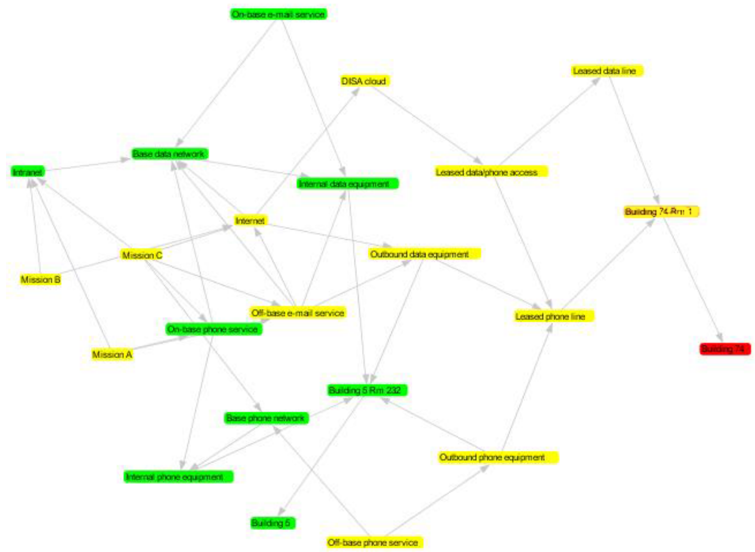

Figure 4.26: This diagram depicts the resultant force directed visualization of dataset 7. The actor designated "Building 74" represents the failed element.

4.8 Dataset 8: AME Mission

The AME Mission dataset consists of thirty-nine actors and a total of ninety-seven direct dependencies. For a complete enumeration of the actors and their dependencies see the dataset xml file in Appendix B.

Because of the sensitive nature of the computer network used for the AME Mission, a simplified network infrastructure has been generated to take its place. This dataset is designed to demonstrate the visualization's ability to model the requirements and interdependency of an actual Air Force mission.

4.8.1 Radial Graph: Dataset 8 - AME Mission. Figure 4.27 depicting the AME Mission dataset further demonstrates that radial visualizations for highly interdependent datasets result in a view that is cluttered and distracting. None of the actors could be selected to produce a view without one or more actors overlapping. The result is degradation of the clarity of the information presented and thus its over all usefulness. This is reflected in the scores in Table 4.11.

4.8.2 Directed Graph: Dataset 8 - AME Mission.

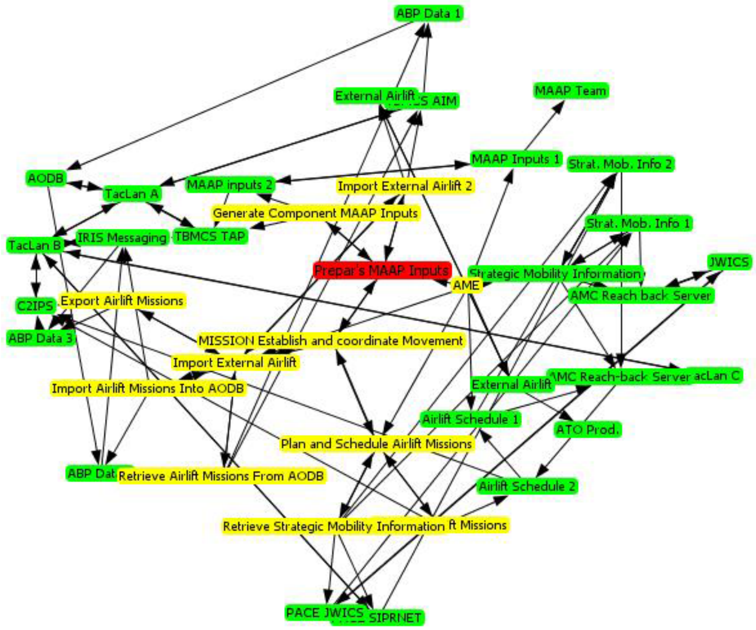

Figure 4.27: This diagram depicts the resultant radial visualization of dataset 8. The actor designated "Prepar's MAAP Inputs" was selected as the focus as it represents the failed element.

Table 4.11: Clarity, scope, and ease of use metrics for the visualizations of Dataset 8

Dataset 8			
	Radial	Directed	Force Directed
Clarity	6	6	9
Scope	8	7	9
Ease of use	8	8	7

4.8.2.1 Clarity. The directed visualization for dataset 8 provides further confirmation that directed visualizations for highly interdependent datasets results in a view that requires extensive manual adjustment. Without extensive adjustment, the visualization is undecipherable in Figure 4.28. Because of this clutter, the overall clarity of the visualization suffers, further degrading the score as can be seen in Table 4.11.

4.8.3 Force Directed Graph: Dataset 8 - AME Mission.

Figure 4.28: This diagram depicts the resultant directed visualization of dataset 8. The actor designated "Prepar's MAAP Inputs" represents the failed element.

4.8.3.1 Clarity. The force directed visualization of the AME Mission dataset presents a good picture of the situation. This as with all of the other datasets, the force settings required adjustments, and there was a few seconds waiting involved for the actors to separate. This result serves to further support of the consistency of the forced directed visualization and resulted in the same scores as before as is shown in Table 4.11.

4.9 Enhancements

Based on the above analysis, this research added enhancements that can be applied to any of the visualizations. The force directed visualization was selected for the purposes of testing and implementing the enhancements. Several general enhancements were applied to improve the visualization's performance in terms of clarity, ease of use, and scope. This research developed and implemented four visualization enhancements. The first is highlighting with color, actors directly impacted

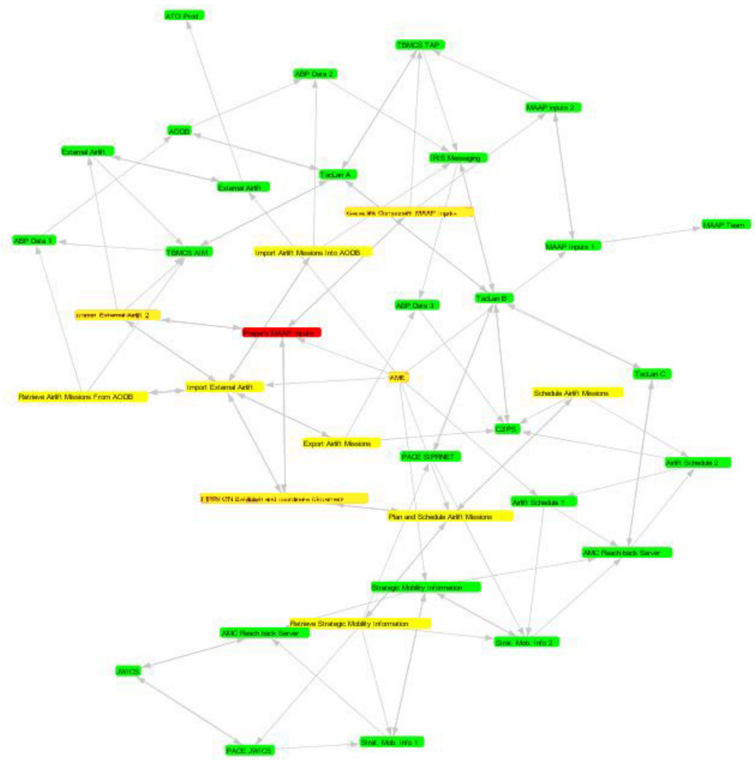

Figure 4.29: This diagram depicts the resultant force directed visualization of dataset 8. The actor designated "Prepar's MAAP Inputs" represents the failed element.

vs. indirectly impacted affected. Second is the ability to display a sub-graph instead of the full graph. Another enhancement is the storing and displaying additional information about each actor. Fourth, this research adds the ability to list every actor that is directly or indirectly dependent on an actor of the user's choosing. Finally, a search function is added enabling users to rapidly located actors.

4.9.1 Directly Impacted. By highlighting the actors directly dependent on an actor that has failed with a different color, improves the clarity and scope of the visualization, see Figure 4.30. This is because the visualization not only makes it clear what needs workarounds to avoid a cascading failure, but it also increases the information a user can gather at a glance. Without such highlighting the user would need to trace the dependency manually.

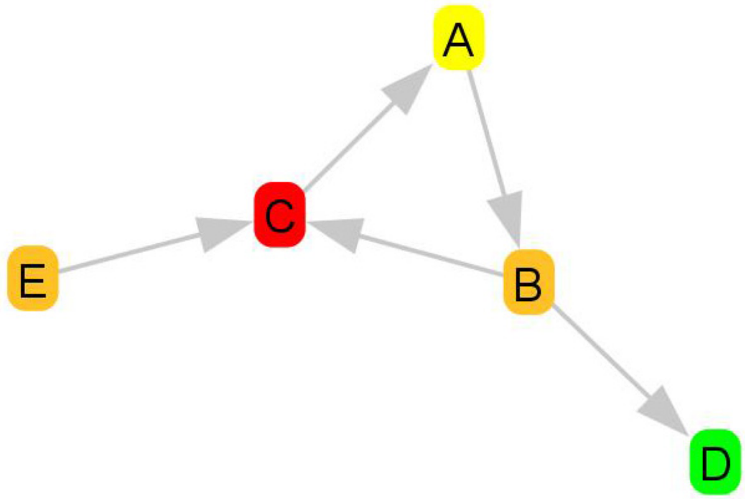

Figure 4.30: This diagram depicts the resultant visualization of using dataset one with direct dependents of the failed element highlighted in orange (E and B). The actor designated C represents the failed element.

4.9.2 Sub-graph. By adding a feature that allows the user to trim away the parts of the visualization the user is not currently interested in and instead focus only on elements dependent on a actor of their choice, the clarity and scope of the visualization is improved as shown in Figure 4.31. This allows a visualization that was once overly crowded, or too big to be easily seen to be trimmed down. Thus offering improved clarity and a scope more useful to the user. When compared to the full graph, shown in Figure 4.26, the added clarity is apparent.

4.9.3 Additional Information. By incorporating additional information into the data structure and displaying it when a user requests it, increases the visualization's scope dramatically without decreasing the clarity of the display. Each actor is displayed using just a single identifying name. Additional information is available for each actor, e.g. location, phone, and point of contact. Loading and storing this metadata in the actor increases the usability for trouble solution. Figure 4.32 shows how the point of contact for an actor can be retrieved. This storing and displaying of metadata decreases response time for resolution.

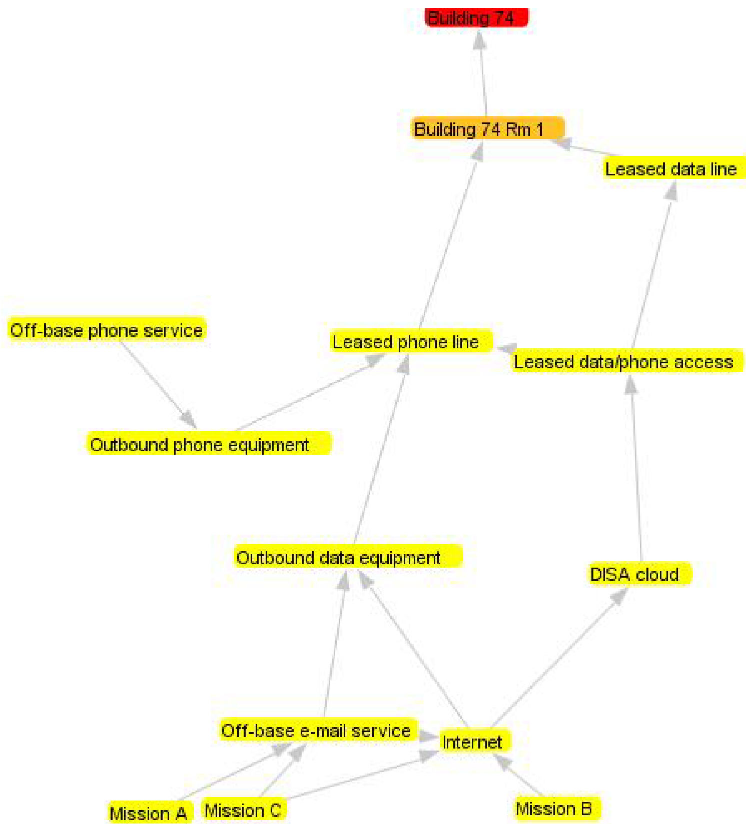

Figure 4.31: This diagram depicts the resultant visualization of using dataset seven to show a sub-graph of only the affected actors. The actor designated Building 74 represents the failed element.

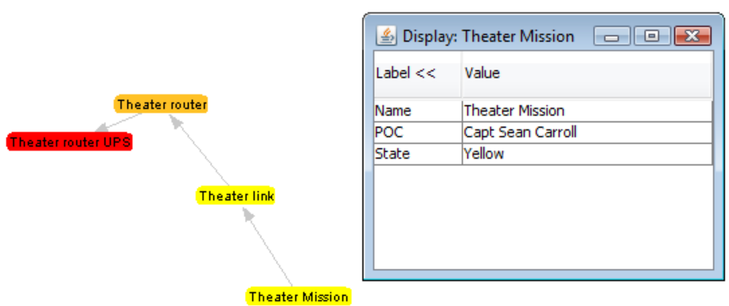

Figure 4.32: This image shows a sub-graph of the affected elements in dataset five and the detailed information for the "Theater Mission" actor

4.9.4 List of Dependent Actors. Taking the concept of additional information a step further, an option to display all of the cascade dependent actors in a table is added. This enhancement allows users to quickly identify exactly who needs to be notified or contacted concerning a situation, as shown in Figure 4.33

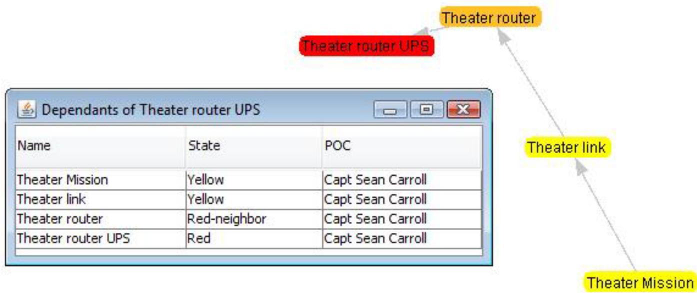

Figure 4.33: This image shows a sub-graph of the affected elements in dataset five and the detailed information for the "Theater router UPS" actor, and all actors dependent on it directly or indirectly

4.9.5 Search Function. The search function enhances the visualizations usability as the number of actors grows. Because of the inclusion of this function it is not necessary for the users to be able to locate a specific actor by eye. Instead a search my be employed to quickly identify the actor of interest. Figure 4.34 shows the results of a search on the term "Base" with the force directed visualization of dataset 5.

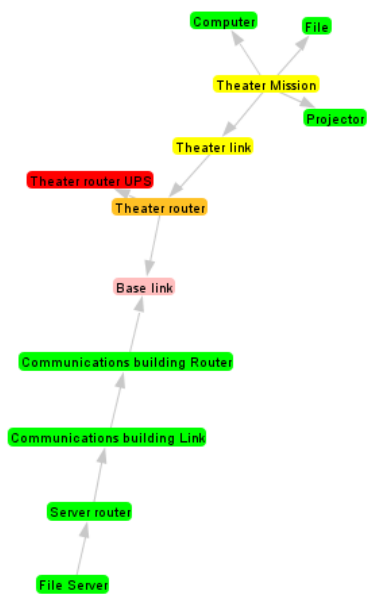

Figure 4.34: This image shows the force directed visualization of dataset five with the search term of "Base".

V. Conclusions

The data discussed and analyzed in Chapter 4 can be correlated to reveal several findings. First, by correlating the visualization analysis based on datasets it shows that the toolkit and process for performing the automated mission impact analysis results in an accurate and useful output. Second, by correlating the analysis of each visualization type, a determination can be made as to the suitability of each visualization. Third, correlating the effectiveness and capabilities of the visualization enhancements it can be shown that many of the problems with the visualization, that would negatively impact its effectiveness, can be overcome. Finally, an analysis of the positives and negatives aspects of different policies for the creation of datasets, demonstrates the flexibility of the approach discussed in this research.

5.1 Findings resulting from the correlation of radial, directed, and force directed graph visualizations in relation to each dataset

By correlating the expected output of the datasets with the actual output of the automated analysis and visualization functions, it is possible to verify that the functions and toolkit is capable of performing a proper analysis on differing dependency structures. To do this, datasets one though four are used to verify the system is capable of handling situations where graph traversal runs from simple to complex. Datasets 5 through 8 are used to verify the handling of more realistic situations.

5.1.1 Datasets 1-4. The purpose of these four datasets is to test the toolkit and the methods created for analyzing the impact of a failed actor. Specifically, as the failure propagates through a simulated system that contains loops, double dependencies, both loops and double dependencies, as well as disconnected graphs. As can be seen in Figures 4.1 through Figures 4.16, in each graph visualization for all four datasets, the nodes representing impacted actors are correctly identified. This success in correctly identifying cascading dependencies indicates that when comparable connectedness is present in a subset of a more complicated situational model, the visualization software will be able to correctly deal with the situation. This ability in turn ensures the dataset may be created as an accurate representation of dependency without the need to simplify the interactions in order to prevent the visualization from failing.

5.1.2 Datasets 5 and 6. These datasets and the visualizations associated with the data demonstrate the software's capability to correctly model a small scale mission. In doing so it further demonstrates that to be useful and effective, a visualization utilizing the actor and need line approach does not need to model an entire base or network. Instead, separate datasets can be built for minor easily mapped missions or a dataset can begin with minor easily identified actors and need lines then expand over time into a more complete picture and be useful from day one.

It can be seen in Figure 4.18, the theater mission dataset, that personnel can recognize the requirements to the theater's mission. They can identify that the theater needs network and file server access to fulfill its mission. With those requirements identified, the NCC can determine what is needed to supply network access to that room or building.

The sixth dataset models the mission needs of a PMI. Though just as easily it could have been any minor mission on a base, such as the requirements of a small tenant unit, or the base travel office. By integrating the smaller disconnected data models into a larger model, a more complete view of the situation emerges. While the data models are in development, visualization continues to function and be of use, since it will operate correctly on disconnected graphs as shown in the Figures 4.15 and 4.16. These disconnected models can then be incorporated into larger datasets and new dependencies defined between them.

5.1.3 Dataset 7. The three visualizations of this dataset as shown in Figures 4.24 through 4.26, demonstrate the ability to extend the visualization well beyond the network to include things such as rooms, and buildings, and by implication other elements that are critical for mission operations such as power and water. Since all possible needs of a mission can be classified as actors and so represented by a node in the visualization, even less tangible requirements can be displayed, for example, adding an actor representing the need for local wind speeds to be below five knots. This element can be updated via base weather or simply used as an indicator of a requirement for simulation purposes.

Dataset 7 also demonstrates the ability to abstract away details that are monitored elsewhere or that would show a level of resolution not desired. In this case, the majority of the base network has been merged into an abstract actor labeled "base data network." This abstraction mandates more work on the part of the personnel responsible for monitoring the visualization in situations where only part of the base data network goes down. Nevertheless, this ability to create and use abstract actors is desirable, as it drastically reduces the number of actors being displayed.

5.1.4 Dataset 8. The ability of the visualization software to correctly identify and display the actors impacted directly and indirectly by a failed actor is shown in Figures 4.27 through 4.29. This fact clearly demonstrates the solutions developed in this research are capable of successfully providing an automated mission impact analysis via comprehensive visualization. Furthermore, with the underlying methodology of dividing everything into the two fundamental categories of actors and need lines, the failure of an actor can be traced though cascading dependencies to all the actors it impacts. This solution is capable of modeling failures beyond the network level, allowing personnel to determine the impact of failing to complete an apparently trivial task on other tasks and the mission itself. This capability is well beyond commercial software currently being marketed for identifying the impact of an outage such as "Whats Up Gold".

Table 5.1: Average clarity, scope, and ease of use for the visualizations of Datasets 1-8

Average Evaluation Scores				
Visualization	Clarity	Scope	Ease of use	Average
Radial	8.00	7.75	8.00	7.92
Directed	8.00	7.63	8.00	7.88
Force Directed	7.00	9.00	9.00	8.33

5.2 *Findings concerning radial, directed and force directed visualizations*

Though the average qualitative score for the radial, directed, and force directed visualizations were nearly identical, varying by less then one on a one to ten scale, the force directed graph would be best suited to most applications. Both radial and directed graph visualizations scored very well in ease of use because there were no controls to manipulate. As a result, just as an axe is easier to use than a chain saw in terms of knowledge needed to operate it both the radial and directed visualizations scored better than the force directed graph. This disparity skewed the cumulative average score resulting in less than a .5 variation between the low score and the high score as is shown in Table 5.1. However, if each category is evaluated separately the force directed visualization out performed the other two in both scope and clarity as is shown in Figure 5.1. Because of this superior performance across two of the three categories the force directed graph visualization is superior even though the difference in the total average score is small.

5.3 *Findings concerning the necessity and functionality of enhancements*

Though the actor and need line system removes the necessity for outages in the resulting visualization to be easily traced to the missions they impact due to the automated process this system allows for it create an over arching difficulty. This result is because the visualization is not divided up into separate visualization layers, as is shown in Figure 3.5 depicting the Multi-layer NCO Model [18]. As a result,

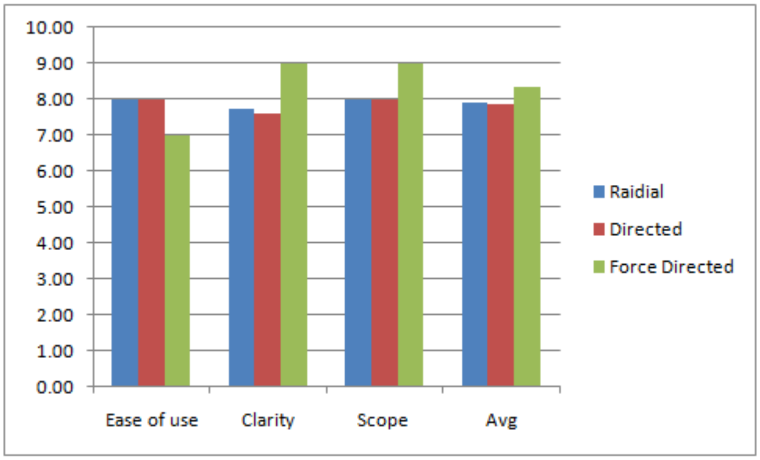

Figure 5.1: This chart provides a graphical representation of the average rating each visualization type received across all dataset and the average of these three scores.

the visualizations created in this research require enhancements to allow the user to quickly utilize the information presented concerning the impact of an event. To this end several enhancements were implemented.

The first of these enhancements is actor details. This allows additional information about every actor to be stored. This information can be elements such as points of contact or brief descriptions. It has been implemented such that additional details can be included in the dataset and the information can be added and labeled without impacting the visualization's functionality. This ensures that as technology matures and policies change, the visualization remains functional and useful, instead of requiring operators to attempt to find a way to identify and add critical information to data formats that are out of date and changing.

The second enhancement implemented concerned highlighting actors that were directly impacted vs. indirectly impacted. This allows a user to identify at a glance, actors directly impacted. When used in conjunction with other enhancements such as actor details and reports, this enhancement can be used to quickly develop workarounds to an equipment outage. For example, if a report, required for a mission to succeed, is normally transmitted via email, the email server would be a direct requirement of the actor representing this report. As a result, if the email server was to go down, the actor representing the report would be identified as directly impacted. The NCC could then call the point of contact for the report explaining the problem and arranging another method for transmitting the document to ensure the success of the mission.

The third enhancement implemented adds the functionality to display a subgraph. This enhancement allows the user to select a single actor and display only the actors and corresponding need lines that are dependent directly or indirectly on the selected actor. This result has the operational impact of allowing someone to focus on only the problem at hand and hide anything not affected by the current situation. This focus enhances users ability to quickly respond to exercise scenarios or questions concerning the potential impact of an outage.

A fourth enhancement is an extension of the subgraph enhancement, providing a textual report detailing the subgraph and information about each actor in it. This shifts the data from a format designed to allow quick visual identification of outages and the scope of the mission impact, to a format that can be used to form an action plan.

The final enhancement implemented has no real impact on the visualization itself. Instead, the search function is designed specifically to aid in locating actors, so that the other enhancements can be used easily. This results in a dramatic change in the overall usability of the visualization, where the system is represented by a large number of actors.

5.4 Future work

This research proves visualization using current technology is viable for mission impact analysis for enhanced situational awareness, and the toolkit used to develop the visualization fulfills the requirements set forth. Further work is needed to develop the visualization to the stage where it can be used by a system integrated across the AF. This future work includes monitoring network and hardware elements, database integration, larger datasets, network integration, and implementation or deployment policy.

Research into monitoring network and hardware elements is currently underway in the via projects related to cybercraft. A direct tie between cybercraft and the visualization could be utilized. Another option would be to develop a data bridge to capitalize on the information gathered by the cybercraft, thus avoiding the complication of a larger project encapsulating both the visualization and cybercraft.

The Perfuse toolkit contains database connectivity. However, these methods and functionalities have not yet been used with this implementation. Linking the visualization to a database would remove the step of formatting the data into xml. Also, according to the Prefuse site, the toolkit has been used for visualizations with very large numbers of elements (thousands to millions). The visualization developed here has not yet been tested or optimized for such large datasets.

Network integration is another area that is in need of further study. The question of the best method to give a large number of individuals access to the visualization at one time? How much of the visualization should individuals have access to? If a mission on base A is dependent on something from base B, should base A have access to base B's information? How inclusive does the data need to be? Should there be one overarching large dataset listing everything possible, or smaller multiple datasets?

Finally, policy concerning an implementation of any large scale Air Force wide program requires a great deal of ground work. Two such areas of the work for this visualization would be to decide what methods for data collection would need to be selected, what would be the classification of the datasets and the resulting visualization.

5.5 Research Impact

Though further work can enhance the usefulness of this research, as it stands the visualization has the potential for immediate significant positive impact. Not only has it shown that a visualization incorporating automated mission impact analysis is possible, it has created such a solution. This solution could be deployed immediately on a voluntary basis. The XML format is simple enough that a unit with no personnel skilled in the Java programming language could still create a viable dataset representing their mission. This dataset could be expanded over months, or even years, giving NCC's and unit control centers an invaluable tool in times of crisis, as well as enhance day to day operations by assisting in the planning and prioritization of repairs or replacements.

Appendix A. Data Set 7: Situation 2 - "Building 74 Destruction" XML file

The following XML file is the file used buy the visualization to represent data set 7. All actors and the detailed information concerning them can be easily located by reviewing the file. As can all dependencies that correspond to the need lines for each actor.

```xml
<?xml version="1.0" encoding="UTF-8"?>

<graphml xmlns="http://graphml.graphdrawing.org/xmlns">
<graph edgedefault="directed">

<!-- data schema -->
<key id="Name" for="node" attr.name="Name" attr.type="string"/>
<key id="State" for="node" attr.name="State" attr.type="string"/>
<key id="POC" for="node" attr.name="POC" attr.type="string"/>

<!-- nodes -->
<node id="1">
 <data key="Name">Mission A</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>

 </node>
<node id="2">
 <data key="Name">Mission B</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="3">
 <data key="Name">Mission C</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="4">
 <data key="Name">Intranet</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="5">
 <data key="Name">Internet</data>
 <data key="State">Red</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="6">
 <data key="Name">On-base e-mail service</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="7">
 <data key="Name">On-base phone service</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
```

</node>
<node id="8">
<data key="Name">Off-base e-mail service</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="9">
<data key="Name">Off-base phone service</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="10">
<data key="Name">Base data network</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="11">
<data key="Name">Base phone network</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="12">
<data key="Name">DISA cloud</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="13">
<data key="Name">Leased data/phone access</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="14">
<data key="Name">Building 5</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="15">
<data key="Name">Building 5 Rm 232</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>
<node id="16">
<data key="Name">Building 74</data>
<data key="State">Green</data>
<data key="POC">Capt Sean Carroll</data>
</node>

```
<node id="17">
 <data key="Name">Building 74 Rm 1</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="18">
 <data key="Name">Leased data line</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="19">
 <data key="Name">Leased phone line</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="20">
 <data key="Name">Outbound phone equipment</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="21">
 <data key="Name">Outbound data equipment</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="22">
 <data key="Name">Internal phone equipment</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>
<node id="23">
 <data key="Name">Internal data equipment</data>
 <data key="State">Green</data>
 <data key="POC">Capt Sean Carroll</data>
</node>


<!-- Junk nodes to force stupid pallet stuff -->
<node id="24">
 <data key="Name">NULL</data>
 <data key="State">Yellow</data>
</node>
<node id="25">
 <data key="Name">NULL</data>
 <data key="State">Red</data>
</node>
```

```
<node id="26">
 <data key="Name">NULL</data>
 <data key="State">Green</data>
</node>



<!-- edges -->
<edge source="1" target="8"></edge>
<edge source="1" target="4"></edge>
<edge source="1" target="7"></edge>

<edge source="2" target="4"></edge>
<edge source="2" target="5"></edge>

<edge source="3" target="7"></edge>
<edge source="3" target="11"></edge>
<edge source="3" target="8"></edge>
<edge source="3" target="4"></edge>
<edge source="3" target="5"></edge>

<edge source="4" target="10"></edge>

<edge source="5" target="10"></edge>
<edge source="5" target="12"></edge>
<edge source="5" target="21"></edge>

<edge source="6" target="10"></edge>
<edge source="6" target="23"></edge>

<edge source="7" target="10"></edge>
<edge source="7" target="22"></edge>

<edge source="8" target="10"></edge>
<edge source="8" target="23"></edge>
<edge source="8" target="21"></edge>
<edge source="8" target="5"></edge>

<edge source="9" target="11"></edge>
<edge source="9" target="20"></edge>

<edge source="10" target="23"></edge>

<edge source="11" target="22"></edge>

<edge source="12" target="13"></edge>
```

```
<edge source="13" target="18"></edge>
<edge source="13" target="19"></edge>

<edge source="15" target="14"></edge>

<edge source="17" target="16"></edge>

<edge source="18" target="17"></edge>

<edge source="19" target="17"></edge>

<edge source="20" target="19"></edge>
<edge source="20" target="15"></edge>

<edge source="21" target="19"></edge>
<edge source="21" target="15"></edge>

<edge source="22" target="15"></edge>

<edge source="23" target="15"></edge>

</graph>
</graphml>
```

Appendix B. Data Set 8 - AME Mission XML file

The following XML file is the file used buy the visualization to represent data set 8. All actors and the detailed information concerning them can be easily located by reviewing the file. As can all dependencies that correspond to the need lines for each actor.

```xml
<?xml version="1.0" encoding="UTF-8"?>

<graphml xmlns="http://graphml.graphdrawing.org/xmlns">
<graph edgedefault="directed">

<!-- data schema -->
<key id="Name" for="node" attr.name="Name" attr.type="string"/>
<key id="State" for="node" attr.name="State" attr.type="string"/>
<key id="POC" for="node" attr.name="POC" attr.type="string"/>
<!-- nodes -->
<node id="1">
 <data key="Name">AME</data>
 <data key="State">Green</data>
 </node>
<node id="2">
 <data key="Name">MAAP Inputs 1</data>
 <data key="State">Green</data>
</node>
<node id="3">
 <data key="Name">Strategic Mobility Information</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="4">
 <data key="Name">MAAP Team</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="5">
 <data key="Name">ATO Prod.</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="6">
 <data key="Name">External Airlift</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="7">
 <data key="Name">Airlift Schedule 1</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>
```

```
</node>
<node id="8">
 <data key="Name">Prepar's MAAP Inputs</data>
 <data key="State">Red</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="9">
 <data key="Name">Import External Airlift</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="10">
 <data key="Name">Plan and Schedule Airlift Missions</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="11">
 <data key="Name">Generate Component MAAP Inputs</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="12">
 <data key="Name">Import External Airlift 2</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="13">
 <data key="Name">Retrieve Airlift Missions From AODB</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="14">
 <data key="Name">Import Airlift Missions Into AODB</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="15">
 <data key="Name">Export Airlift Missions</data>
```

```
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="16">
 <data key="Name">Schedule Airlift Missions</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="17">
 <data key="Name">Retrieve Strategic Mobility Information</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="18">
 <data key="Name">AMC Reach back Server</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="19">
 <data key="Name">MAAP inputs 2</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="20">
 <data key="Name">Strat. Mob. Info 1</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="21">
 <data key="Name">TBMCS TAP</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="22">
 <data key="Name">TBMCS AIM</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
```

```
&lt;node id="23"&gt;
 &lt;data key="Name"&gt;AODB&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="24"&gt;
 &lt;data key="Name"&gt;IRIS Messaging&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="25"&gt;
 &lt;data key="Name"&gt;C2IPS&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="26"&gt;
 &lt;data key="Name"&gt;PACE SIPRNET&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="27"&gt;
 &lt;data key="Name"&gt;PACE JWICS&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="28"&gt;
 &lt;data key="Name"&gt;JWICS&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;

&lt;/node&gt;
&lt;node id="29"&gt;
 &lt;data key="Name"&gt;External Airlift&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
  &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;
&lt;/node&gt;

&lt;node id="30"&gt;
 &lt;data key="Name"&gt;ABP Data 1&lt;/data&gt;
 &lt;data key="State"&gt;Green&lt;/data&gt;
 &lt;data key="POC"&gt; Capt Sean Carroll&lt;/data&gt;
```

```
</node>
<node id="31">
 <data key="Name">ABP Data 2</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="32">
 <data key="Name">ABP Data 3</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="33">
 <data key="Name">Strat. Mob. Info 2</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="34">
 <data key="Name">AMC Reach-back Server</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="35">
 <data key="Name">Airlift Schedule 2</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="36">
 <data key="Name">TacLan A</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="37">
 <data key="Name">TacLan B</data>
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="38">
 <data key="Name">TacLan C</data>
```

```
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>
<node id="39">
 <data key="Name">MISSION Establish and coordinate Movement</data
 <data key="State">Green</data>
 <data key="POC"> Capt Sean Carroll</data>

</node>

<node id="40">
 <data key="Name">NULL</data>
 <data key="State">Yellow</data>
</node>
<node id="41">
 <data key="Name">NULL</data>
 <data key="State">Red</data>
</node>
<node id="42">
 <data key="Name">NULL</data>
 <data key="State">Green</data>

</node>



<!-- edges 1-->
<edge source="1" target="2"></edge>
<edge source="1" target="3"></edge>
<edge source="1" target="6"></edge>
<edge source="1" target="7"></edge>
<edge source="1" target="8"></edge>
<edge source="1" target="9"></edge>
<edge source="1" target="10"></edge>

<!-- edges 2-->
<edge source="2" target="4"></edge>
<edge source="2" target="19"></edge>


<!-- edges 3-->
<edge source="3" target="33"></edge>
<edge source="3" target="20"></edge>
<edge source="3" target="18"></edge>
<edge source="3" target="34"></edge>
```

```
<!-- edges 6-->
<edge source="6" target="5"></edge>
<edge source="6" target="29"></edge>

<!-- edges 7-->
<edge source="7" target="34"></edge>
<edge source="7" target="33"></edge>

<!-- edges 8-->
<edge source="8" target="39"></edge>
<edge source="8" target="11"></edge>
<edge source="8" target="12"></edge>

<!-- edges 9-->
<edge source="9" target="39"></edge>
<edge source="9" target="12"></edge>
<edge source="9" target="13"></edge>
<edge source="9" target="14"></edge>
<edge source="9" target="15"></edge>

<!-- edges 10-->
<edge source="10" target="39"></edge>
<edge source="10" target="16"></edge>
<edge source="10" target="17"></edge>

<!-- edges 11-->
<edge source="11" target="8"></edge>
<edge source="11" target="19"></edge>
<edge source="11" target="21"></edge>

<!-- edges 12-->
<edge source="12" target="8"></edge>
<edge source="12" target="9"></edge>
<edge source="12" target="29"></edge>
```

```
<edge source="12" target="22"></edge>

<!-- edges 13-->
<edge source="13" target="9"></edge>
<edge source="13" target="22"></edge>
<edge source="13" target="30"></edge>

<!-- edges 14-->
<edge source="14" target="9"></edge>
<edge source="14" target="24"></edge>
<edge source="14" target="31"></edge>

<!-- edges 15-->
<edge source="15" target="9"></edge>
<edge source="15" target="25"></edge>
<edge source="15" target="32"></edge>

<!-- edges 16-->
<edge source="16" target="10"></edge>
<edge source="16" target="25"></edge>
<edge source="16" target="35"></edge>

<!-- edges 17-->
<edge source="17" target="10"></edge>
<edge source="17" target="26"></edge>
<edge source="17" target="27"></edge>
<edge source="17" target="33"></edge>
<edge source="17" target="20"></edge>

<!-- edges 18-->
<edge source="18" target="28"></edge>

<!-- edges 19-->
<edge source="19" target="21"></edge>
<edge source="19" target="2"></edge>
```

```
<!-- edges 20-->
<edge source="20" target="3"></edge>
<edge source="20" target="18"></edge>



<!-- edges 21-->
<edge source="21" target="24"></edge>
<edge source="21" target="36"></edge>

<!-- edges 22-->
<edge source="22" target="30"></edge>
<edge source="22" target="36"></edge>

<!-- edges 23-->
<edge source="23" target="31"></edge>
<edge source="23" target="36"></edge>

<!-- edges 24-->
<edge source="24" target="32"></edge>
<edge source="24" target="37"></edge>


<!-- edges 25-->
<edge source="25" target="37"></edge>

<!-- edges 26-->
<edge source="26" target="33"></edge>
<edge source="26" target="37"></edge>


<!-- edges 27-->
<edge source="27" target="20"></edge>
<edge source="27" target="28"></edge>


<!-- edges 28-->
<edge source="28" target="18"></edge>
<edge source="28" target="27"></edge>


<!-- edges 29-->
<edge source="29" target="6"></edge>
```

```
<edge source="29" target="22"></edge>


<!-- edges 30-->
<edge source="30" target="23"></edge>

<!-- edges 31-->
<edge source="31" target="24"></edge>


<!-- edges 32-->
<edge source="32" target="25"></edge>


<!-- edges 33-->
<edge source="33" target="3"></edge>
<edge source="33" target="34"></edge>

<!-- edges 34-->
<edge source="34" target="38"></edge>
<edge source="34" target="35"></edge>


<!-- edges 35-->
<edge source="35" target="25"></edge>
<edge source="35" target="7"></edge>

<!-- edges 36-->
<edge source="36" target="21"></edge>
<edge source="36" target="22"></edge>
<edge source="36" target="23"></edge>
<edge source="36" target="37"></edge>

<!-- edges 37-->
<edge source="37" target="24"></edge>
<edge source="37" target="25"></edge>
<edge source="37" target="26"></edge>
<edge source="37" target="36"></edge>
<edge source="37" target="38"></edge>

<!-- edges 38-->
<edge source="38" target="34"></edge>
<edge source="38" target="37"></edge>
```

```
<!-- edges 39-->
<edge source="39" target="8"></edge>
<edge source="39" target="9"></edge>
<edge source="39" target="10"></edge>

</graph>
</graphml>
```

Bibliography

1. "Jgraph Website". URL `http://www.jgraph.com/`.
2. "Whats Up gold Website", January 2008. URL `http://www.whatsupgold.com/products/whatsup gold premium/`.
3. Barnes, Piet, Josh; Hut. "A Hierarchical O(NlogN) Force-Calculation Algorithm". *Nature*, 324:446–449, Dec 1986.
4. Chen, Xiao Fan Wang Guanrong. "Complex networks: small-world, scale-free and beyond". *Circuits and Systems Magazine, IEEE*, 3(1):6–20, Jan-Mar 2003.
5. Garstka, John J. "Network Centric Warfare: An Overview of Emerging Theory". *Phalanx Online*, 33(4):1–33, Dec 2000.
6. Goldschmidt, David. *Design and Implementation of a Generic Graph Container in JAVA*. Master's thesis, Rensselaer Polytechnic Institute, April 1998.
7. Hearst, Ka-Ping Yee Danyel Fisher Rachna Dhamija Marti. "Animated Exploration of Dynamic Graphs with Radial Layout". *IEEE Symposium on Information Visualization*. 2001.
8. Keller, A. Formella J. "Generalized Fisheye Views of Graphs". *Graph Drawing*, 242–253. September 1995.
9. Landay, Jeffrey Heer Stuart K. Card James A. "Prefuse: a toolkit for interactive information visualization". *Conference On Human Factors in Computing Systems; CHI 2005*. 2005.
10. Minden, D. L. Tennenhouse J.M. Smith W.D. Sincoskie D.J. Wetherall G. J. "A Survey of Active Network Research". *Communications Magazine, IEEE*, 35(1):80–86, January 1997.
11. Money, Arthur L. *Report on Network Centric Warfare Sense of the Report*. Report to the congress, section 934 of the defense authorization act for fy01, US Goverment, March 2001.
12. Nicholas Economides, Evangelos Katsamakas. "Linux vs. Windows: A comparison of application and platform innovation incentives for open source and proprietary software platforms". The Networks, Electronic Commerce, and Telecommunications (NET) Institute, September 2005.
13. Shaw, Alfred K. *A Model For Performing Mission Impact Analysis Of Network Outages*. Master's thesis, Air Force Institute of Technology, March 2007.
14. Stanley, Jeffrey E. *Enabling Network Centric Warfare Through Operational Impact Analysis Automation*. Master's thesis, Air Force Institute of Technology, March 2005.

15. Stanley, R. F. Raines R. A. Baldwin R. O., J. E. Mills. "Correlating network services with operational mission impact". *MILCOM 2005. IEEE*, volume 1, 162–168. IEEE, 2005.

16. Stein, David S. Alberts John J. Garstka Frederick P. *NETWORK CENTRIC WARFARE: Developing and Leveraging Information Superiority*. DoD C4ISR Cooperative Research Program, 2nd edition (revised) edition, 1999.

17. Winograd, Doantam Phan Ling Xiao Ron Yeh Pat Hanrahan Terry. "Flow Map Layot". *Infovis*. 2005.

18. Wong-Jiru, Ann. *Graph Theoretical Analysis of Network Centric Operations Using Multi-Layer Models*. Master's thesis, Air Force Institute of Technology, September 2006.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
27-03-2008	Master's Thesis	Aug 2006- Mar 2008

4. TITLE AND SUBTITLE
Mission Impact Analysis Visualization for Enhanced Situational Awareness

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Carroll, Sean, C. M., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GCO/ENG/08-01

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
THIS SPACE INTENTIONALLY LEFT BLANK

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
N/A

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This research effort examines the creation of mission impact analysis visualizations to enhance situational awareness. It focuses on using prefuse to create a visualization that allows the user to quickly understand the impact of the failure of any element needed directly or indirectly for a mission. The visualization correctly identifies the direct or indirect impact on physical requirements such as network links and servers as well as non-physical elements such as the generation of a report, or ability to perform a task. The visualization provides an overview of the situation, as well as including enhancements to allow for greater detail on any element to be viewed. The result of this research is the foundation for a tool to allow commanders and others, at a glance, to understand the scope of mission impact when an outage occurs.

15. SUBJECT TERMS
Visualization, Mission Impact Analysis, Mission Impact, Situational Awareness, Prefuse, Actor, Need Line

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stuart H. Kurkowski, Lt Col, USAF (ENG)
U	U	U	UU	114	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-3636 x7228 Stuart.Kurkowski@afit.edu

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.