Spring 5-2011

A Methodology to Develop a Communication Protocol for Visualizing Simulations in a Collaborative Virtual Reality Environment

Lacey Suzanne Duckworth
University of Southern Mississippi

The University of Southern Mississippi

A METHODOLOGY TO DEVELOP A COMMUNICATION PROTOCOL FOR VISUALIZING SIMULAITONS IN A COLLABORATIVE VIRTUAL REALITY ENVIRONMENT

by

Lacey Suzanne Duckworth

Abstract of a Dissertation
Submitted to the Graduate School
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

May 2011

ABSTRACT

A METHODOLOGY TO DEVELOP A COMMUNICATION PROTOCOL FOR

VISUALIZING SIMULAITONS IN A COLLABORATIVE

VIRTUAL REALITY ENVIRONMENT

by Lacey Suzanne Duckworth

May 2011

In the technology field, simulations and collaborative virtual reality environments (CVREs) are not generally combined because it is complicated to develop large scale simulations within CVREs. The complexity of combining these two technologies in order to form a better form of visualization stems from the lack of a methodology to help derive these scalable simulations. Simulations require very complex calculations that the CVRE cannot perform as it is overloaded in calculations for the maintenance and stability of the environment itself. Since the simulation cannot be held within the CVRE, the solution is to move the simulation external to the CVRE and provide means for the CVRE and simulation to communicate so the scene within the CVRE can be updated. While this increases the performance of the simulation in the CVRE, another element is required to make the simulation scalable. Since the CVRE controls the interactions and the simulation controls the calculations and reactions, the basic structure of the this operations can be visualized as a state machine. By implementing the simulation as a state machine, if another element needs to be added to the simulation, it is a matter of implementing a new state and adding the transitions between the new state and all pre-existing states. Implementing the simulation as a state machine leaves the CVRE responsible for the visualization of the simulation and provides means for the simulation

and CVRE to communicate, which leads to the idea of a new developmental methodology for the visualization of large scale simulations in CVRE. This methodology will result in the ability to provide simulations in need of a visualization to be quickly and cost effectively implemented in a CVRE so that single users can visualize and interact. This methodology will not only impact those in need of simulations in the result of more simulation and training software, but also provide a better workforce equipped with decision-making tools and more widely available simulation and training software.

The University of Southern Mississippi

A METHODOLOGY TO DEVELOP A COMMUNICATION PROTOCOL FOR VISUALIZING SIMULATIONS IN A COLLABORATIVE VIRTUAL REALITY ENVIRONMENT

by

Lacey Suzanne Duckworth

A Dissertation
Submitted to the Graduate School
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

Approved:

Dr. Andrew Strelzoff ____________________
Director

Dr. Tulio Sulbaran ____________________

Dr. Joe Zhang ____________________

Dr. Ray Seyfarth ____________________

Dr. Nan Wang ____________________

Dr. Susan A. Siltanen ____________________
Dean of the Graduate School

May 2011

ACKNOWLEDGMENTS

The writer would like to thank the dissertation chair, Dr.Andrew Strelzoff, co-chair, Dr. Tulio Sulbaran, and the other committee members, Dr. Zhang, Dr. Seyfarth, and Dr. Wang, for their guidance and encouragement throughout the duration of my graduate career. I would like to extend a thank you outside the realm of The University of Southern Mississippi to Dr. Randy Rapp of Purdue University and Professor Hal Johnston of Cal Poly University both for providing me with a test case for this dissertation. Last, but not least, this dissertation would not be possible without guidance from my parents' and fiancé, and the miracles God has performed throughout my life.

Special thanks go to Dr. Strelzoff and Dr. Sulbaran as they helped me learn the importance of organization, timeliness, and challenged me so that I will be prepared for the workforce. I am very thankful that God placed me under their guidance as they are wonderful advisors, but most of all friends.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF ILLUSTRATIONS

Figure

CHAPTER I

INTRODUCTION

Client-Server Virtual Reality Environments that are intended for social networking are attractive platforms to develop simulations as they reach millions of people world-wide with an accessible, compelling, and engaging visual and audible experience. However, these environments were not intended for large scale simulations. Limitations in the internal, event driven language for virtual environments force programmers into difficult and un-reusable design patterns. Using this approach to program large scale virtual simulations results in long segments of code spread among multiple objects, effectively, "spaghetti" code. External to the virtual environment, a variety of specialized simulation languages exist which are easy to use and provide strong re-usability of code designs. The realization of the importance for addressing the problems of developing a simulation in a CVRE came from the following experiences:

1. Research where a simulation was needed for a refinery process in a CVRE
2. A test to calculate the run time of a small equation inside a simulation language versus inside a CVRE

Simmulation for Refinery Process

The first realization that such a communication protocol was needed for developing simulations in a CVRE was when the virtual reality research group, under direction of Dr. Andrew Strelzoff at The University of Southern Mississippi, tried simulating one process at the Chevron Oil Refinery Plant of Pascagoula, Mississippi in a CVRE. The consultant provided the diagram (Figure 1) of the refinery process the plant to be simulated.

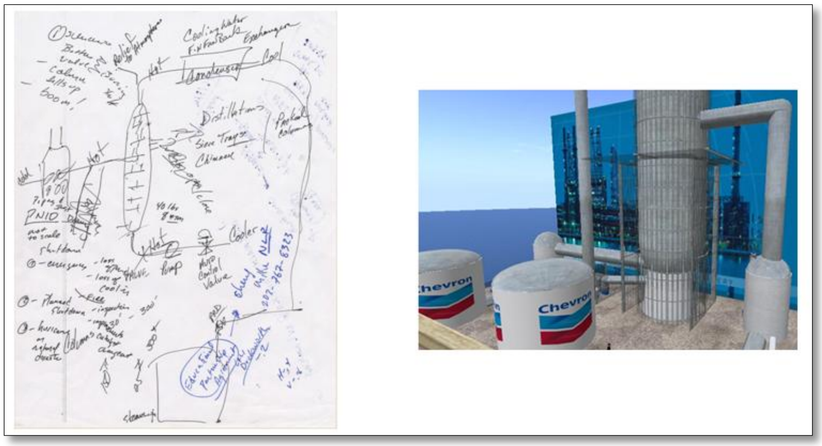


Figure 1. Consultant provided diagram (left) and CVRE interpretation (right).

Creating this process, which was only one of the forty six processes, took 200 hours to complete. If the same method was used to implement the remainder of the refinery process, the man hours required to fully simulate the entire Pascagoula Chevron Plant, where each process increases on a scale of difficulty by ten, would be 96,000 hours. This study provided insight as to why client-server collaborative virtual reality environments were never intended for large scale software development as three contributors were determined to cause the time complexity for of such simulations. The first contributor was that SecondLife and all similar client-server VR environments were never inteded for large-scale software development. An ideal solution for this would be developing an object-oriented language for “top-level” process rigs where all instances and specialized rigs could be easily derived. The second contributor is that computer scientists are not refinery plant operators which made it very hard to “get it right” from conversations and schematics provided by the oil refinery experts. In order to address this contributor, an ideal solution would be to develop a high-level behavioral simulation environment in which programmers would contribute a first version and plant operators and consultants could iteratively “get the details right” largely on their own. The last contributing factor is that the client-server virtual reality environments were never intended for large scale computations. As the number of numerical processes increase, the servers tend to bog down and cause performance degrades. An ideal solution would be to have an external simulation language that could run independently on a dedicated server providing scalable performance as the size of the simulation grew. In addition, this would prevent “spaghetti” code from occurring as there would be a methodical means for designing and implementing the code.

Comparing Run Time of a Small Equation in Simulation Languages Versus CVRE

The second realization, which led to insight as to why large scale simulations were not supported in CVREs, as stated in the previous section, was when a test was done to determine how long it would take to run a simple calculation in a CVRE versus a simulation language. This simple calculation was to sum all of the odd numbers between one and one million and display the results. In this test case, SecondLife was selected as the CVRE which uses the Linden Scripting Language (LSL), a programming language specifically for SecondLife (SecondLife, 2008). LSL is similar to JAVA and C/C++ but does not have the object oriented features. In fairness to the code structure of SecondLife, the simulation programming langue used was C++ and LabVIEW. The idea for using C++ and LabVIEW was to have a comparison between the runtime of LSL versus a language it was based on and a comparison between a simulation language such as LabVIEW. The table below shows the average of three run times for each language.

Table 1

Run-Time of Simulation Languages Versus CVRE

Language	Average run time of calculation in seconds
LSL	24.708785
C++	0.0148
LabVIEW	0.3314

This table shows that the LSL language took the longest time to run such a simple calculation. On the other hand, C++, the language from which LSL was based took about 24.693985 seconds less to run and the simulation language, LabVIEW, took only 24.377385. The rational for LSL taking so long to compute this equation is that most of the computing time is spent updating the environment in the CVRE. Although it takes a little longer time than C++, LabVIEW still runs the calculation in a shorter time than the LSL language. This lead to the idea that moving the calculation external to the CVRE and only using the CVRE would provide a robust and scalable communication protocol.

In order to address this problem, the objective of this dissertation was to develop a robust and scalable communication method that connects external simulation languages with client-server Virtual Reality Environments. The desired results of this dissertation were a model for developing a communication protocol to connect a simulation language to a VRE that will allow more complex and scalable simulations to be developed faster. This would amplify the needs for simulations in VREs in areas such as construction, medical technology, education, as well as various other fields and extend existing simulations which might not benefit from a visual simulation due to its complexity.

CHAPTER II

REVIEW OF RELATED LITERATURE

One of the fastest growing areas in the field of computer science is simulation through virtual reality environments. The interest in this area is due to its ability to help employers train their employees, the desire to improve gaming graphics and interactivity, and various other reasons. In order to illustrate the existing state of simulation within virtual reality environments (VREs), this chapter provides an overview of Virtual Reality, Simulation, and Communication Protocol.

Simulation

A simulation is a method for representing key characteristics or behaviors of a physical or abstract system (Wikipedia, 2011). Simulations are beneficial for testing various products or methods as they help learn skills in an interactive way, minimize risks associated with decision making, problem solving and strategy implementation. Many companies are using simulations to help train their employees, visualize results quicker, and determine the outcome of a choice selection in order to help save time and money. As simulations prove to be beneficial the demand for different verities of simulations increases. This section gives an overview of some of these projects including plant life simulation for the gaming industry, crowd simulation, smoke simulation, and construction equipment and process simulation.

Plant Life Simulation

Julie Dorsey, a Computer Science Professor at the University of Yale, as well as some colleges and graduate students developed a simulation to realistically render botanical structures. This simulation is useful as it can be used in performing environmental assessments and developing accurate graphics for gamming where accurate botanical graphics are desired. Dorsey and her group worked together and developed a real-time plant rendering software package using OpenGL. In order to accurately render the plants, a multiphase process was developed. First, a bidirectional reflectance distribution functions (BRDF) and bidirectional transmittance distribution functions (BTDF) were used to develop images. Two of each of these functions had to be derived in order to represent both the top and bottom of a leaf. The BRDF functions generated the following maps: albedo, specular intensity, and the specular roughness map each of which were developed through equations that manipulated the image pixel by pixel. However, for the BTDFs, only the thickness detail map is generated. Upon generating the maps, several other functions are applied to the leaves including sunlight decomposition, light-visibility convolution, compression, and the level of detail. Applying a combination of these functions determines the alterations made to the leaves or plant. In order to apply these various functions, a graphical user interface was developed with a description of each window (Dorsey, Wang, Wang, Yang, Guo & Shum, 2006).

Upon completing the rendering simulation, it was compared with two other methods. Although not clear from the figure, the contribution showed that as long as measured data is available such as the plant leave thickness, and various other attributes, a significantly simpler appearance model can be derived without deteriorating the quality of the image. In other words, the interior of the material does not have to be modeled in order to predict the appearance (Dorsey et al., 2006).

Crowd Simulation

In order to understand the flow of pedestrians, urban crowds, armies, vehicle traffic, and animal groups such as flocks, herds, and schools in the environment, crowd simulation is used. It is important to understand how these object move in various environments and under certain conditions. For example, it would be useful to ensure that if a fire happened in a certain location of a building what would be the maximum number of individuals that could safely exit before serious damage occurred. In order to simulate the movement of large groups of autonomous characters in real time, an average of 60 frames/sec, Craig Reynolds and his colleges at Sony Computer Entertainment US Research and Development used the powerful PlayStation 3 (PS3) cell architecture to develop a crowd simulation. The components of a PS3 include six Synergistic Processor Units (SPU) with 256 Kbytes of memory each (Reynolds, 2006).

The algorithm developed to manage the crowd simulation was based on a particle system where each individual in a crowd is represented by a particle having a behavior and the ability to interact within its environment which runs in O(n) time. However, because particles in this case can interact with other particles, the time increases to O(n2).

In order to increase the run time associated with crowd simulation, a spatial subdivision technique was used. In this technique the spaces occupied by a space that an individual could occupy are divided into smaller regions and each individual is associated with one of these local regions. To implement these techniques, a “bucket” method was developed and instantiated into a C++ Library. The components of this library include an individual, a container class, which involves a bucket, lattice, and NearestN, and a bucket update parameter.

In this environment, an individual is one member of a crowd, the container class is contains a collection of individuals in a occupying a particular space, the update parameter reassigns individuals to a bucket when it crosses into a different buckets boundary, a lattice class is a 3D array of identical buckets, and the NearestN maintains the nearest neighbor of an individual. In order to make use of the PS3 hardware the SPUs maintain the crowd simulation by updating the buckets in parallel and disjoint sets. It is the Data Memory Access that instances the data to the Reality Synthesizer Graphics Processing Unit, the PS3s graphic chip. This simulation worked according to design and presented the performance times.

Smoke Simulation

One of the most difficult objects to simulate in computer graphic applications is smoke due to the time complexity associated with computing the complex equations required to accurately simulate smoke. In order to reduce the time complexity, Ronald Fedkiw and his colleges from Stanford University used Euler equations which are less computationally intensive than the popular Navier-Stokes equation. Another problem associated with smoke simulation was that it was too slow or suffered more numerical dissipation. To eliminate this, Fedkiw and his colleges introduced a smie-Lagrangian integrating scheme as well as a pressure-Poisson equation to create a stable model. By combining these two methods, they implemented a “vorticity confinement” algorithm which would take the energy from the numerical dissipation and inject it back into the equation using a forcing term to increase the vorticity of the flow. In other words, once a smoke was started, the equation would keep the smoke alive and flowing in the correct direction for a period of time (Fedkiw, Stam, & Jensen, 2001).

Construction Simulation

In the field of construction, simulation can save time and money as employees can be trained and construction sites modeled. By using simulators to train employees, equipment costs, damage, and time away from a job site is eliminated, the safety of the employee is not in jeopardy, and there is an instructor available for feedback. On the other hand, construction projects simulated are useful because they can take into account possible obstacles, ensure the plan is planned out appropriately, as well as assure that all materials are present. In the remainder of this paper the following will be discussed: simulation of construction equipment and a construction process disturbance simulation analysis tool.

Simulation of Construction Equipment

In order to eliminate the problems associated with training employees with actual equipment, The Southwest Research Institute developed the Excavator Operator Training Simulator (EOTS) in cooperation with a major manufacturer of construction equipment. TSWRI created a software package that operates on a personal computer allowing an instructor to teach the skills associated with operating the machinery and trainees the ability to practice in a risk-free environment eliminating the chances of equipment or personal injuries. In order to describe the complexity of the system designed, it is easiest to look into the depth of material covered in the tutorial. The tutorial includes the following topics: maneuvering and positioning the excavator, digging staring and level trenches, loading trucks effectively, moving and placing loads such as trench boxes and pipe, and loading the excavator onto a transport trailer. In the training mode, the trainee is faced with potential safety hazards including operations near power lines, open trenches, other equipment, facilities, and personal, that must be responded to before he/she can complete the lesson. In addition to these obstacles other standard operation rules are applied to the equipment. For example, all equipment has a load limit that must be maintained in order to ensure the equipment does not tip over. If the trainee crosses this limit, in practice mode a warning would be given. However, if the trainee makes mistakes in the test mode, it affects his performance score. In calculating this score, a unique approach is used to help the employer determine if the employee should work with this equipment. The score is determined based on the trainee's productivity against the labor and equipment cost incurred, which includes the damage done to the equipment by the trainee as well as the rent rate (Fisher, 2008).

Simulation of Construction Disturbances

There are many aspects that can affect the progress of a construction project. Some possible interferences include the weather conditions, soil conditions, available space, traffic, the qualifications of employee personal, and the reliability of subcontractors. Because none of these interferences can be predicted a simulation tool that would consider how a combination of these possibilities would affect construction progress is beneficial. Fritz Gehbaeur and his colleges at the Lean Management Institute developed a simulation based analysis of disturbances in construction operations. First, the team developed a disturbance database which contained the construction project, disturbance registration, disturbance eliminations, and disturbance effects. The construction project simply defined what was to be done at the construction site. The disturbance registration registers an actual occurrence of a disturbance including the time it occurred, duration, and type of disturbance. The disturbance elimination stored the time frame for clearing the problem as well as the requirements for addressing the situation. Lastly, the distance effects where the financial and physical state of the employees, bystanders, equipment and material.

Before simulating interferences, the operator must develop an initial throughput diagram for the construction process as seen in Figure 10. Basically, the contractor outlines the construction project as it would happen if an interference did not occur. Afterwards, the GUI can be used to apply different interferences either before a phase, after a phase or in the midst of a phase. In other words, the possibilities of testing some part of a construction phase against interferences are endless (Gehbaeur et al., 2007).

As seen from the examples previously discussed, simulation covers a variety of applications from gamming, strategy planning, and training. While being used as a strategy tool for construction project planning or equipment training both time and money can be optimized. On the other hand, using simulation to perfect moving images and perform plant rendering can increase the intensity of games, movies, as well as help scientist perform environmental assessments.

Virtual Reality

One of the areas technology continues to branch into is Virtual Reality. Perfecting this field is at the center of attention because its finished product helps various audiences either through interactions with a 3D environment which allow for manipulation and exploration of various worlds. Virtual reality targets audiences ranging from young children, the working class, professionals, and various other individuals as it can provide entertainment, training, or a combination of both to in order to make learning fun. As these virtual environments are in high demand a lot of time and money is spent by programmers in order to design these virtual environments. This section gives an overview of virtual reality including its definition and characteristics, components, some examples of current virtual reality applications, and problems programmers face when developing virtual environments.

Definition

The most commonly accepted definition for a virtual environment is an environment representing a 3D world and interaction is provided so the world can be manipulated and explored so that the user feels immersed within the world (Brooks, 1999; Strickland, 2008). While in the world, the user represented as an avatar can either be isolated by themselves, a virtual reality environment or with multiple users, a collaborative virtual reality environment (CVRE). In both types of virtual environments, two aspects have to be considered when trying to immerse the user in the virtual environment which includes the breadth and depth of information.

Breadth of information relates to the number of sensory dimensions the user undergoes. In a virtual environment, the sensory dimensions include, sound, visual, and touch in increasing order of importance. If at any point, the user encounters a poor breadth of information, then the sense of immersion is interrupted causing the user to leave the virtual environment. An example of this would be if a user is in a virtual world where the trees are still and they feel a gust of wind. Being that the feeling of the wind does not relate to the visual image, the user tends to step out of the virtual world to check on reality (Strickland, 2008).

Depth of Information relates to the quality of the data the user receives when interacting within the environment, particularly the quality of the image and audio output.

Within a virtual world, graphics have to be on a lifelike scale to provide visual immersion and displayed at a fast enough resolution in order to minimize latency. For example, if a user was looking at a cow standing 10 feet away from them, the cow would not be 10 inches tall. Also, if the cow mooed, it would not be a very low pitch sound. In other words, when designing a virtual environment, a good sound and visual location has to be relevant to the user's location. In addition to location, in regards to the sound and visual of an image in virtual reality, if the user alternates the image in any form, the scene should change immediately reflecting this alteration. In most virtual realities, frames are updated around 20 – 30 frames/second which allows the visual aspect of immersion to be satisfied (Brooks, 1999; Strickland, 2008).

In order to improve the breadth and depth of immersion in virtual environments the base software from which the environments are developed are upgraded. Beginning in the 1980s, OpenGL was one of the popular virtual environments for visualizing simulations but it only processed data to make a visual and could not be interacted with by the user. Later in 1996, Java 3D was developed which allowed a virtual environment to show the results of a simulation and the user could tweak the values in the simulation through interaction but still could not visualize themselves in the environment as an avatar. It was in 1999 that SecondLife made it possible to interact with objects and others as an avatar in a CVRE. While some simulations could be performed in Second Life there were limitations to its size as discussed in Chapter I of this dissertation.

Components

Regardless of what virtual environment selected, several components are required to keep the sense of immersion. These components can be classified into several categories including the following: visual display, aural devices, graphics rendering system, a tracking system, and some means of data storage for maintaining the virtual world (Brooks, 1999; Strickland, 2008).

Various types of displays exist for exhibiting virtual worlds including, head mounted units, CAVEs, panoramic projectors, and desktop displays. Although included in the list, most do not include desktop displays by posing the argument that user's cannot be completely immersed in a virtual world while looking at such a small screen. However, many popular games exist in the form of computer and console units requiring the user to use some sort of desktop display unless they are able to afford some of the more expensive forms of displays. With a few more dollars, head mounted units and panoramic projectors can be purchased to provide a better form of immersion for an individual. The head mounted units are similar to a pair of goggles the user wears to encapsulate them by watching a 3D movie. On the other hand, panoramic projectors are a larger scale of the head mounted units as they project the 3D environment to a user wearing a cheaper set of goggles. Perhaps the most expensive display is the CAVE, which was developed at the Electronic Visualization Laboratory at University of Illinois at Chicago, creates a virtual environment while using 10 x 8 foot walls as projecting screens, which project images from high resolution LCD projectors (Chan, 2006).

Examples of Virtual Reality

As mentioned earlier, there are several virtual environments available for entertainment, training, research, and various other purposes. In the entertainment realm, virtual reality can be experienced through console games such as the Nintendo Wii TM, computer games, 3D movies, and roller coasters. While not all roller coasters have the virtual reality aspect, Disneyworld's Cyber Space Mountain allows a user to personalize their own roller coaster, sit in a CAVE like environment and enjoy their creation. Computer games include the popular World of Warcraft and Sims where users take on a role of a character and perform survival and trivial activities in order to progress through a life like game. Some job fields involved in virtual reality include architects, interior designers, construction crews, and the military.

In the field of architect and interior design, the contractor can design a house/ room respectively as they believe the home buyer/owner envisions. Once this creation is complete, the buyer/owner can take a virtual walk through to see if the intent was conveyed and allows the buyer/owner to determine if this is what they really want as an end result.

On the other hand, in the field of construction virtual reality is used to create simulators to provide adequate training on assembly line processes, machine operation, and various other tasks which prevent the damage of expensive materials and catastrophic events from occurring (Strickland, 2008). Chans crane simulator, was implemented by clustering several inexpensive computer systems in a high-speed interaction communication network in order to take advantage of distributive virtual reality.

In addition to training simulators, construction professionals also use simulators to help clients explore the characteristics and time-line of building construction (Brana, 2003). In Brana's thesis work, he developed a method for importing Gant Projects into VRML and showing the status of the building at certain intervals which can be manipulated by the user. This gives both the construction personnel and the client an idea of how the building should progress as well as a end result visual to ensure the client will be happy with the end result, assuming the work is completed as described.

Another field that also uses virtual reality for training purposes is the military. The trainee steps into a simulator device and feels as if he or she is stepping into an actual tank, plane, or helicopter. The trainees also have the ability to connect with other military personnel several miles away via Internet connection. These personnel include first-responder units, civilians, or medical personnel units providing a more realistic form of training as all of these units depend and interact with each other during actual combat. While operating in the virtual environment, the trainees use personally designed avatars so they personally do not have to step through dangerous places such as a checkpoint in Baghdad, or a subway in New York under a chemical attack by terrorists (Wilson, 2008). In addition to using virtual reality to train individuals, the military also uses virtual reality as a treatment for post traumatic stress disorder (PTSD) for war veterans. When treating veterans to this form of therapy, they are placed in virtual environments where they experience the same traumatic event which caused them to have PTSD via a video clip. In addition to the visual aspect, veterans are also exposed to acoustic startles, which make the experience even more realistic. Studies show that those who undergo this form of treatment as well as a complimenting medication to help speed the process have a higher success rate of conquering their fears (NIMH, 2008).

One of the most popular uses for virtual environments is by incorporating the aspect of Collaborative virtual environments with education. Monahan and team members developed a Collaborative Learning Environment (CLEV-R) which was a web-based 3D environment where multiple users could experience real-time teaching with assistant from the professor and other students without encountering the issues of peer-pressure due to appearance or learning curves (Monahan, McArdle, & Bertolotto, 2008). In CLEV-R the traditional school rooms included a lecture room, meeting room, and library. While the lecture room and library were for public use, the meeting room could be utilized for private conversations. While several tests were performed to determine the benefits of CLEV-R most of the results were to be expected except the degree of immersion. Only 78% of the students who were used to test the environment experienced immersion (Monahan et al., 2008). While this could be as a result of numerous contributors such as frame rate, calculation rate, connection speed, etc., the author did not specify why these results resulted.

Conclusion

While this section discusses some of the applications of simulations in virtual environments several other fields use this technique for training including Civil Engineering, medical, crowd simulation, and various others (Basdogan, Ho & Srinivasan, 2002, National Institute of Mental Health, 2008, Reynolds, 2006 & Sampaio, Henriques, & Martins, 2010). Virtual reality allows users to escape reality and enter a world where their actions can render several reactions. With virtual reality, entertainment and education can be covered either simultaneously or consecutively as many use it as a form of recess while others use it to disguise work. In order to keep the attention of users by sustaining immersion, programmers must ensure both the breadth and depth of information is efficient, which can cost the programmer several hours depending on the complexity of the desired virtual environment.

Communication Protocol

A communication protocol is a set of rules for the representation and transformation of data over a communication channel(s). When dealing with communication protocols, much effort is spent trying to optimize the algorithm used for the communication process in order to minimize the amount of time required to send, receive, and signify proper message received flags. In addition to optimization, many communication protocols are added rapidly in order to provide a new means of communication between the server and client. This section discusses several communication protocols such as a LEGO/Handy Board communication protocol enhancement, an enhanced BlueTooth radio communication protocol, an overlay network communication protocol, a decentralized architecture for virtual environments, high-performance VR design, and a grid based environment.

Lego/Handy Board Communication Protocol Enhancement

LEGO robots are driven by a Handy Board controller(s) which are very useful as long as the application does not involve a high level connection capability. In this project Dakai Zhu and Ali Tosum from the Department of Computer Science at the University of Texas at San Antonio combine the LEGO Handy Board with a Tmote Sky sensor to enhance the communication of the LEGO robot through use of radio-frequency communication capability. The MIT Handy Board, was based on the Motorola MC68HC11 micro-processor with a rich set of functionalities, and has a simple interface for connecting with other LEGO robot components. One of its strongest traits is its ability to connect with other boards of its same type. Although the Handy Board has an IR device for communication, it is not an optimal source as it requires the robots to be within a short distance from each other and direct pointing is required. In order to circumvent this, Zhu and his team used the Tmote Sky sensor to provide wireless communication through an on-board antenna, allowing the robots to be within 125m (approximately 410.10 feet) maintaining stable communication. The Tmote Sky sensor connects to the Handy Board through use of the RX, TX, and ground pins within the USB. By connecting these pins, both the Handy Board and Tmote Sky can be programmed so that LEGO robots can communicate with others through the radio frequency communication provided by the Tmote Sky sensor in more reasonable distances from each other (Zhu & Tosun, 2008).

Enhanced Bluetooth Radio Communication Protocol

In technology, there are many devices that communicate via Bluetooth, one of the current popular wireless communication protocols. Although Bluetooth devices are cost efficient, they suffer from low power and short-range communication distances. Some key aspects of Bluetooth communication include piconets, bridges, and scatternets. It is important to first understand these concepts in order to understand how a two layer multicast communication protocol, developed in collaborative effort between the Department of Information Management at Hsiuping Institute of Technology and the Department of Computer Science and Information Engineering at Tamkang University.

One of the most popular communication protocols used in Bluetooth communication is a piconet, which is a group of Bluetooth devices where one Bluetooth acts as the master device maintaining communication for seven other slave devices. In the piconet, a master stays in an inquiry state obtaining information about the other slaves within its network. The information is stored in a packet and contains the slave's address and clock information used to derive hop sequences. Once a master has obtained this information, in other words, completed the inquiry stage, it moves into a page state where the master sends a slave a frequency hop synchronization packet containing a 3-bit active member address. Now a communication link is created between the master and all slaves.

Another element used in piconets is a bridge, which is one node that relays messages between different piconets. This node is unique as it can take on one of the modes when acting between two sets of piconets, slave/slave or master slave. If the node is acting in a slave/slave mode, it plays the slave mode in alternating piconets. On the other hand, if acting as a master slave, one node is a master and the other is a slave.

One of the last components involved in Bluetooth communication is a scatternet, which is a multicast service to perform data transmissions from some source within the piconet to several other members that are outside the piconet of the message sender. In order for a multicast service to perform, an efficient multicast tree must be constructed so that the source of the sender can reach all other members. There are three specific features that a multicast tree must possess. First, the tree must contain as few as possible nodes that are not within its piconet, in order to minimize the band-width consumption and the number of broadcasts. Secondly, the height of the tree must be at a minimal in order to reduce the delay associated with all members receiving the message. Lastly, the tree must contain as many shared links as possible in order to reduce once again the band-width consumption. Other than the restrictions above, the tree is constructed with the source as the parent node and all other nodes connecting to it as a child.

In order to enhance the existing Bluetooth communication protocol, Chang and his college developed a two-layered multicast communication protocol that applies the role switching technique to aid in constructing an efficient multicast tree. The first thing Chang and his group did was enhanced the construction of the multicast tree by including the characteristic of role and hopping sequence within a Bluetooth network. To handle this, the source of the message became the root and each piconet was forced to have only seven member slaves in order to maintain the height restriction. All internal nodes where the members are required to join two piconets would play the master/slave role in order to assure message delivery to all piconets (Chang, Chang, & Chang, 2008).

In order to construct the multicast tree, a flooding mechanism was used. When a message is sent from a master, each bridge switches to another piconet and forwards it to the other devices within that particular piconet. The flooding mechanism helps ensure that all members get the message quicker than if they all waited to receive the message from the source and is maintained by the Two-layer Multicast Communication Protocol (TMCP), which consist of two phases.

In the first phase, the multicast tree is constructed by using the flooding mechanism in order to send the initialize packets and assign roles to each member. Here, the source enters the page scan state and waits for responses from other members, where the member and non-member nodes of a piconet have different responsibilities. The job of the non-members is to forwards search packets to all of its neighbor nodes by extracting the message sent by the source, replacing it with its information and sending it down the line. All other members stay in the page scan mode until time has begun for phase 2.

In phase 2, the role switching operations are applied to the protocol in order to establish the topology of the constructed multicast tree. In order to do this, two masters from different piconets swap connections with the slave by having the slave broadcast a query packet to its children sending a packet asking them to make contact with the source.

The TMCP is used to construct a tree for a multicast Bluetooth network where the main idea is to take some given scatternet, treat it is as the first-layer of communication, and develop a multicast tree as a second layer. This protocol ensures that the power, bandwidth, and propagation delays within a Bluetooth communication are at a minimal.

An Overlay Network Communication Protocol

With malicious individuals watching traffic over the internet, information is not always safe. Due to this problem, much effort has been applied to increasing internet privacy. The traditional method of protecting ones information is encryption. While encryption does conceal one's name, password, etc., it fails in hiding the users IP address and email address if choosing an electronic delivery option. In order to address this problem, Pan Wang and his colleges from the Department of Electrical and Computer Engineering at North Carolina State University, developed a simple and scalable anonymous communication protocol in order to provide a k-anonymity to the sender and receiver during internet communication (Wang, Ning, & Reeves 2007).

In this type of communication protocol, there are k number of nodes participating in a network where k can take on any value from 1 to N. While participating in a network, all nodes are indistinguishable and organized into a set of logical rings in order to form an overlay network over the internet. In order to provide the anonymity between two nodes, message batching and a one-way key-chain is used in order to ensure nodes in the network can send message to all other nodes within its ring maintaining anonymity, and preventing them from modifying or replacing a message with malicious data. (Wang et al., 2007)

When participating in a network, a node takes on one particular mode of a super node or general node. A super node is one having the ability to connect with nodes outside its private network and a general node is one that can only connect to other nodes within its network. In order for a message to be sent from a node in one particular network to a node outside of its network, the message is originally sent to a randomly chosen agent node, or super node, in its local ring.

Once the agent node receives the message, it is extracted and forwarded to the corresponding destination ring. This ensures the receiver does not know who the actual sender of the message was. On the other hand, if a node wishes to send a message within its network, more work is required. Like external network communication, a batch message is created but extended to include one message from each node in the ring. Each node encrypts their message and ID. The starting from a source, each node forms a unique one-way key chain to each of the other nodes in the network. The key chain is formed by starting with some super node in the ring that forms an initialization packet containing a dummy message and an n-slot key vector to contain a secret session ID and a message authentication code. When passing around, each node grabs one of the vector slots and one message batch increasing by one until all nodes have now can communicate in the same manner as above.

A Decentralized Architecture for Virtual Environments

Another utilization of communication protocols is in the development of virtual environments. Solipsis was developed by Frey and Rovan where they attempted to solve the issue of emergence of large scale virtual worlds by utilizing peer-to-peer technology. Solipsis was built to manage multi-participant, user-generated virtual worlds by distributing the communication and computational cost among various nodes within a virtual space. Defining their environment as a metaverse consisting of avatars, objects, and sites (locations to be occupied by objects or avatars), each item in this metaverse was distributed over a set of hosts that maintain information about all entities within the metaverse. In order to keep up with all information in the metaverse each node would contain in an entity descriptor a list of hosts that contain information associated with the 3D description. While a direct recipe was not provided, Frey and Rovan gave an illustration to show the different layers and association between objects, avatars, or sites in the three layers of the physical network, raynet layer for determining closest neighbor, and the 3D space (Frey & Rovan, 2008).

High-Performance VR Design

In order to provide a recipe for creating virtual reality applications Allard and his team attempted to associate a data-flow model with a hierarchical component model. Part of developing this model was as a result of algorithmic issues in the calculations of large scale simulations, software engineering design in regards to integration, and the hardware performance limitations. The model developed was FlowVR which included a set of distributed tasks that obtain data from input ports, use the data to make calculations or determinations, and send the results through an output port. The method for processing data is handled using the first-in-first-out (FIFO) algorithm. While the FIFO method seems like a bad choice at first, Allard and his team correct the negative aspect of this algorithm by quickly distributing the tasks to other target machines to perform the calculations. Using this scheme, a hierarchy design and data flow graph was developed that could be utilized for designing other simulations. When testing the model, Allard and his team realized that unless users were FlowVR experts they would not be able to use this method, the communication between processors could only be done with C++, as well as other drawbacks to the design (Allard, Lessage, & Raffin, 2010).

A Grid Based Environment

Like Allard and his teams' idea to improve algorithmic cost, a different method was utilized to implement a simulation in hopes of creating visualization by improving the computing efficiency by utilizing grid-based technology (Allard et al., 2010). The idea of a collaborative virtual geographic environment (CVGE) for the planning of silt dam systems was implemented by Hui and his team. In order to apply grid-based technology to his design, Hui and his team used MA which was a set of autonomous software that could travel amongst hosts within a heterogeneous network in order to interact with other agents or share resources in hopes of calculating the results needed to visualize the simulation. Utilizing Java for displaying their scene, C++ and OpenGL for fast terrain rendering, java media framework for chat, and Voyager for mobile agent computing they were able to produce distributed applications. Not only were they successful in implementing a simulation using this methodology, they also provided a generic model for describing how to divide services between the mobile agents and the logic design of their simulation (Hui, Jun, Gong, Bingli, & Hua, 2010).

Conclusion

As seen in this section, communication protocols can be as simple as two different devices communicating between one another or as complex as multiple agents working together to calculate a calculation to determine a visualization. Without communication protocols systems would have to do the work on their own which would require a substantial amount of wear and tear on the system and prevent the development of large scale simulations as discussed in this section.

CHAPTER III

RESEARCH METHODOLOGY

To help develop the communication methodology for the visualization of simulations in Collaborative Virtual Reality Environments (CVRE) a research methodology was developed. The methodology for this dissertation included two stages, where each finished stage allowed the progress toward deriving the generalizable and scalable communication methodology which was the focus of this dissertation.

Stages

In this dissertation, three stages were developed to help in the derivation of the generalizable and scalable communication protocol which included:

1. Selecting and implementing a concrete example;
2. Defining the generalize communication protocol between an external simulation language and a client-server CVRE; and
3. Testing the robustness and scalability of the proposed design using a test case.

The purpose for selecting and implementing a concrete case, stage 1, was to help derive the generalized communication protocol between the simulation languages and CVREs. Using the information derived in stage one, stage twos purpose was to develop a mapping between the finite state machine definitions of the generalized simulation language($\Sigma 1, S1, s01, \delta 1, F1$) and the event-driven state machine of the client-server VRE ($\Sigma 2, S2, s02, \delta 2, F2$) which would result in a communication protocol methodology. Once the hypothesized methodology was derived stage three was used to test the communication methodology developed in stage 2 using a different test case to determine if the hypothesized model worked as hypothesized.

In order to determine what was needed completing at each stage of the methodology, each stage was categorized as a particular type of research by using "Practical Research Planning and Design" by Paul Leedy & Jeanne Ormrod. Based on the knowledge from this book, stage two was categorized as a "Qualitative – Content Analysis" and stage 1 and 3 as a "Qualitative – Case Study" (Leedy & Ormrod, 2009).

The definition and guidelines for a Qualitative - Content Analysis as explained by Paul Leedy and Jeanne Ormrod is one that encompasses several approaches to research where each approach has two things in common focusing on phenomena occurring in the "real world" and studying the entire complexity of that phenomenon. Content Analysis is a detailed and systematic examination of the contents of a particular body of material for the purpose of identifying patterns, themes, or biases. Table 1 provides the steps used to complete the second stage of this dissertation where the left column is the steps proposed by Leedy and Ormrod (Leedy & Ormrod, 2009).

Table 2

Methodology for Completing Stage Two

Steps in Qualitative Content Analysis Research	Implementations in this Research
1. Identify the specific body of material to be studied in its entirety	The following body of knowledge will be studied: a. Virtual Reality c. Network Technology b. Simulation, and
2. Define the characteristic or qualities to be examined in precise, concrete terms.	1 Virtual Reality will be examined as a state machine that encompasses states (1a), alphabet (1b) and responses (1c) which will be defined. 2 Simulation will be examined as a state machine that encompasses states, alphabet and responses. Specific states(2a), alphabet(2b) and responses(2c) will be defined. Mixed, Discrete and Continuous simulation types are expected. 3 The properties of Network Technology will be examined to determine the best choice to propagate events and responses between VR and simulation state machines.
3. Break down each item into small manageable segments that are analyzed separately	1 The Virtual Reality states (1a) will be categorized (ex: 1a1. touch 1a2. General applications). The VR alphabets (1b) will be categorized following a change of state inputs. The VR responses (1c) will be categorized using the organization and documentation provided by LindenLab as a starting point and then addition extensions will be included (Ex: Shadows from realXTend) . 2 The Simulation states (1a) will be categorized (ex: 1a1. touch 1a2. General applications). The Simulation alphabets (1b) will be categorized following a change of state inputs. The Simulation responses (1c) will be categorized using the organization and documentation provided by LindenLab as a starting point and then addition extensions will be included (Ex: Shadows from realXTend) . 3 Necessary services to propagate events and responses between the VR and simulation using Network Technology will be examined. Examples are "wait_for_read", "opne socket", etc.
4. Scrutinize the material for instances of each characteristic or quality defined in step 2.	1 For each of the Virtual Reality states (1a1..1an), scrutinize and identify the accepting alphabets (1b1.. 1bn) and resulting responses (1c1..1cn). This will lead to a comprehensive set of "skeleton adapters" which represent an organization of the code needed to move the VR state machine across the network to the "Simulation side". 2 For each of the Simulation states (2a1..2an), scrutinize and identify the accepting alphabets (2b1.. 2bn) and resulting responses (2c1..2cn). This will lead to a comprehensive set of "skeleton adapters" which represent an organization of the code needed to move the VR state machine across the network to the "VR side". 3 For each required network service the necessary inputs and output and relationship among services will be determined resulting in a block diagram level skeleton implementation of the networking for our protocol including the response adapter for the VR objects.

The methodology used for the first and third stage, "Qualitative - Case Study", can be defined using the previous definition of Qualitative paired with the definition of a Case

Study where something is studied in depth for a defined period of time. A Qualitative – Case Study is a form of Qualitative research where in-depth data is gathered relative to the topic for the purpose of learning more about the unknown or poorly understood situation. Table 3 shows the various steps of stage 2 as related to Qualitative-Case Study where the right column indicates the steps necessary to complete stage 2 and left column contains the steps as proposed by Leedy and Ormrod (Leedy et al., 2009).

Table 3

Methodology for Completing the First and Third stage

Steps in Qualitative Case Study Research	Implementations in this Research
1. Organization of details about the case – specific facts about the case are arranged in a logical order	Stage 1- This step involves selecting and describing a motivational large scale simulation problem Stage 3 - This step involves selecting and describing a motivational simulation problem
2. Categorization of data – categories are identified that can help cluster the data into meaningful groups	Stage 1 and 3The data to be gathered can be divided into the following categories: Size, complexity and performance of Simulation-driven VR, and the development time as the Simulation-driven VR scales to large sized and increases in complexity relevant to the chosen motivating example.
3. Interpretation of single instances – specific documents, occurrences, and other bits of data are examined for the specific meanings they might have in relation to the case	Stage 1 - Design VR, Simulation and Network Technologies Stage 3 - Design VR, Simulation and Network Technologies using skeletons provided in stage 2 – specific to the chosen motivating example in 1.
4. Identification of patterns – The data and their interpretations are scrutinized for underlying themes and other patterns that characterize the case more broadly than a single piece of information.	Stage 1 - Test the performance of the developed communication protocol Stage 3 - Test our thesis using the case study. Does this methodology allow larger, more complex Simulation-driven VR environments (with good performance) to be built more rapidly?
5. Synthesis and generalizations -An overall portrait of the case is constructed. Conclusions are drawn that may have implications beyond the specific case that has been studied.	Stage 1 and 3 - Write up of results which would include time tables of run time versus complexity, build time versus complexity, etc.

The purpose for classifying each stage as a particular type of research methodology was to help determine the steps for completing this dissertation, and thus became the milestones for the dissertation. In the process of completing each milestone, a detailed list of tasks was developed.

Detailed Methodology

While the main stages for this dissertation were determined early, the actual steps required to complete the dissertation were not clear until the work began. The following paragraphs discuss the steps that were required for completing each stage of this dissertation which included:

- Stage 1: Selecting and implementing a concrete example;
- Stage 2: Defining the generalizable communication protocol between an external simulation language and a client-server CVRE; and
- Stage 3: Testing the robustness and scalability of the proposed design using a test case.

Stage 1: Selecting and Implementing a Concrete Example

In order to select a concrete example that would help derive the desired communication protocol and be of benefit to a client, some research was done to determine what fields simulations in CVREs were not as developed. After researching this area, it was apparent that the construction field had very few simulations in CVREs because the calculations for developing construction based simulations are too rigorous for the CVRE to handle. The lack of being able to implement these types of simulations in CVREs was a good problem for this dissertation as the goal was to develop a communication protocol that would allow the calculations for the simulation to be external to the CVRE. Once the field of the test case was determined, the next step was to determine an actual construction simulation to implement in the CVRE.

Contact was made with Dr. Randy Rapp, business relations of dissertation co-chair, Dr. Tulio Sulbaran, who was interested in having an excavation training simulator for students in his course, Building Construction Management, at Purdue University. The purpose of this simulation was to help students visualize their choice of excavation method based on the Occupational Safety and Health Administration (OSHA) standards. Thus, the basic idea became to provide a means for students to answer questions which determined the type of excavation to be made, checking these answers and then providing a visualization of the simulation which was an excavator excavating the particular site.

After the concrete example was selected, it was time to implement the simulation in the CVRE. The first step in implementing the code required deciding what languages to use to develop the simulation and the CVRE. After some research, LabView was selected to implement the simulation and RealXtend was selected for visualizing the simulation in the CVRE. From this point, the code to visualize this exercise in the CVRE was first implemented in LabView and then in RealXtend. Once the code was complete, it was time to test the design by letting Dr. Rapp's students experiment with the software.

Preparing the excavation training simulator for testing required creating a server to host the simulation in the CVRE, developing usernames and passwords so students could log into the CVRE, a tutorial for introducing the software, surveys to determine if the simulation was usable and motivational, and organizing a test date with student volunteers from Dr. Rapp's class. The students were given one day from 8:00 a.m. until 5:00 p.m. CST to log into the CVRE, complete as many exercises as they wished and then answer the two surveys. Once this day was over, the survey results and the server logs were collected in order to determine the success of the excavation training simulator. Despite the results of the server logs and the survey results, the next step was to look at how the simulation in the CVRE was implemented and determine the generalizable communication protocol.

Stage 2: Defining the Generalizable Communication Protocol

In order to determine the components of the generalizable communication protocol, the implemented code was studied in depth to determine its basic structure. It was noticed that the code in the simulation language, LabVIEW, resembled that of a state machine where input, in the form of a state was received from the CVRE a state transitioned to, and the results returned to the CVRE. Thus, the next step was to map the components of the simulation code onto a state machine and add two additional components for receiving and sending states from the CVRE. Once this mapping was determined, the CVRE was analyzed to determine the main components required to execute any simulation. The basic components were determined to be an object to trigger a state, an object to simulate the results calculated in the simulation language, a slave object for aiding in completing the simulation and components for communicating with the simulation language. Once all of the components were determined a complete diagram was developed to illustrate the communication protocol and a methodology was developed to state the order which element should be implemented. Once the hypothesized communication protocol was determined, the next step was to verify that this design worked for any simulation and any CVRE.

Stage 3: Testing the Robustness and Scalability of the Proposed Design

To determine if the hypothesized communication protocol worked required selecting a test case. In addition to utilizing this test case to verify the accuracy of the communication protocol, the test case also had to test the robustness and scalability of the communication protocol. Based on the given test bench, a more complicated simulation was needed to implement in the CVRE. This resulted in pairing with another business relations of Dr. Sulbaran, Hal Johnston, Construction Management Professor at Cal Poly University. Mr. Johnston is owner of the Building Industry Game, BIG, which is a computer simulation of a realistic business environment where participants play the role of contractors, competing in a market with variable demand for construction work. Since the game was text based, the idea was to develop a means to help contractors visualize their bids to make sure they are selecting the correct one. This visualization would be where they selected a method for excavating and then presented a visual before confirming their bid. The first step was to take the hypothesized communication protocol and determine the components required to implement the desired simulation. Once this was done, the next step was to obtain the BIG code from Mr. Johnston.

The code received did not have any documentation so the next step was to determine how the code worked. This was done by tracing the database while manipulating the simulation by interacting with its web front end. Determining the workings of BIG allowed more details to be completed on the communication protocol diagram. Once the BIG code was understood, the next step was to determine what languages to use for the simulation and the CVRE.

Because BIG was written using Java Server Pages (JSP) the simulation language selected was PHP and the CVRE was selected to be O3D, a web based CVRE. After selecting the languages, the communicaiton protcol methodology derived in stage 2 was used to design the simulation visualizaiton of the excavation method selected by the student. After the design was complete, the code to handle the information received from Simulation Language and the code for visualizing the simulation in CVRE was implemented in BIG and O3D respectively. Once the simulation was complete, a meeting was held with Professor Johnston where he was given a tutorial and then provided independent testing time. The information from this meeting and the experience form implementing this simulation using the communicaiton protocol methodology was used to modify the methodology.

CHAPTER IV

CONCRETE CASE STUDY

A concrete case study was selected in order to establish the requirements for a generalizable communication protocol between a simulation language and collaborative virtual reality environment (CVRE). This chapter provides an overview of the selected case study, preliminary design, code implementation, client and server deployment, tests against several performance, development and usability metrics and the lessons learned.

Selected Case Study

The case study selected was a project with Dr. Randy Rapp, Construction Associate Professor Purdue University. Dr. Rapp requested a training simulator to teach students in his class the appropriate way to excavate a construction site in compliance with OSHA standards. Students would practice determining the appropriate way to excavate a site using the following information:

1. The project bid plan; and
2. Soil type information for the project site.

This information would be used to answer four questions related to the properties of excavation relative to its given properties and obtaining immediate feedback. The current method to execute similar exercises was using paper-and-pencil to teach students about excavations. This simulation, unlike the paper-and-pencil pedagogy would differ in presenting a dynamic environment where calculations and exercise logic were located in a server and visualization was presented in the Collaborative Virtual Reality Environment. The ability to provide a training simulator for the students was just one of the many reasons this case study was selected. Other reasons for selecting this case study included:

1. A training simulator fitting these requirements did not exist;
2. Training simulators such as this are in demand for teaching the appropriate construction methods and a consequence of mal-practicing constructions methods; and
3. It was an application of real world software engineering as a client, Dr. Rapp, provided some specified requirements to be implemented.

Training Simulators of This Type Do Not Exist

Although CVRE training simulators are not new in the field of technology, they are predominantly used in the medical field, aiding trainees preparing to perform surgical operations and lack in the construction field due to the complexity of the desired simulations. Similar to the medical field, the construction field could benefit greatly from CVRE training simulators so that construction operations can be mastered. Construction CVRE simulators are few in number because implementing them can be very difficult for computer scientists as there are many rules governing a construction project.

The Demand for Training Simulators Like This

The current method for teaching students to properly excavate a construction site was, as mentioned earlier, a series of paper and pencil exercises. This process may be insufficient for some students as the two-dimensional aspect of paper is an admittedly limited representation of a multidimensional activity. Simulated hands-on experience aids students' comprehension of multi-dimensional activities and obtains immediate results based on their actions. Ensuring that students understand the appropriate way to excavate is vital to their future success as completing a construction project without flaws is costly, but costs are even greater if the student fails to properly execute a step in the construction process.

Similarity to Real World Application of Software Engineering

Dr. Rapp provided specifications necessary to design and implement the excavation training simulator. From these specifications, the project would be designed, implemented, tested, and reconfigured, the same process as real world software engineering. In the next section, the setup and execution of the details of the exercise are explained.

While some specifications of the project were provided about the case study prior to its selection, more was needed in order to implement the simulator. Dr. Rapp provided a word document entitled “Excavation Decision-making Flowchart for Virtual Reality Lesson” and a power point slideshow entitled “Production Tables from Nunally and Others.” The word document provided the general outline of the simulation: information that a student would be given, information students were expected to know, and the calculations associated with excavations. The power point slideshow included highlights of the word document as well as key terms used in construction excavations.

While this information was beneficial in providing the basic details of the case study, more information about construction excavation was needed in order to fully understand how an excavation was made. Dr. Tulio Sulbaran, a construction professor at The University of Southern Mississippi, gave an extended lecture on excavations and provided *Construction Professional Reference* (2005), a reference book by the Dewalt

Trade Reference Series (Dewalt, 2005). This book provided necessary information such as a complete list of all soil types, the slope methods for each of the soil types, and the slope ratio for the combinations of each soil type and slope method.

The flow chart seen in Figure 2 was desinged in order to demonstrate the excavation process and the case study to be implemented, a flow chart was prepared. The flow chart as seen below not only illustrates the excavation process, but also demonstrates how the students' answers to each question affect the results. As seen in Figure 2, the two main components that influence the students' answers would be the project bid plan and the soil type. An example of a bid plan and soil type can be seen in the Figure 3.

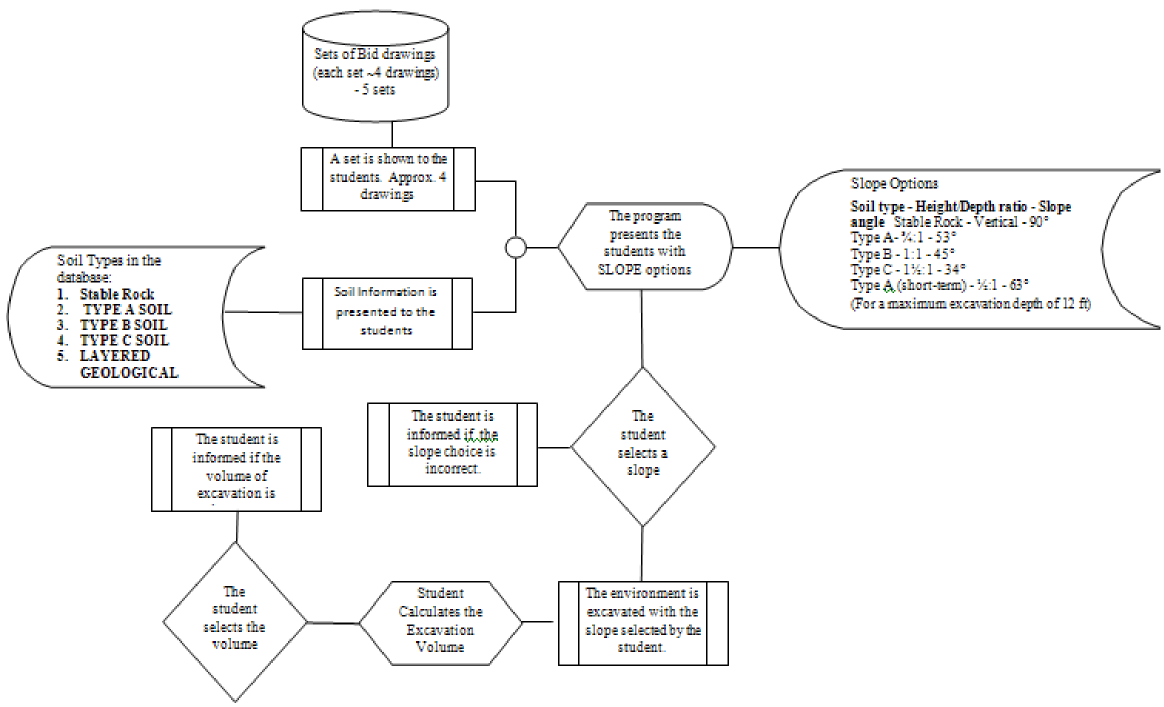


Figure 2. Flowchart of case study.

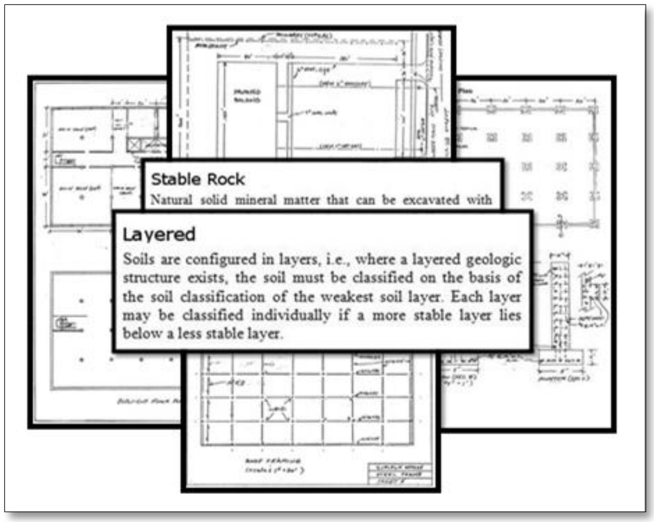


Figure 3. An example of two soil types (center) and four bid plans (exterior).

While there are numerous types of bid plans and soil types, this particular exercise used only three sets of bid plans as well as three soil types as per Dr. Rapp's request. The multiple choice questions to be answered by the students included:

1. Which soil type does the exercise best describe;
2. What is the appropriate slope method for the soil type provided;
3. What is the appropriate height/depth ratio for the soil type and slope method selected; and
4. What is the correct volume of excavation.

Upon choosing their answer to each question, a student would obtain two forms of visual feedback. The first form of feedback was a display window which shows the choices selected by the student and a notification message indicating why their selection was correct/incorrect. A notification message provided the means to give feedback to the students. For example, if the student, when given soil Type A was asked to determine the appropriate slope method for the soil type provided, selected a Simple Slope, the display message would state: "You selected the correct slope method for type A soil." The correct slope ratio for Type A soil is ¾:1. The second form of visual feedback is an excavator in the CVRE that digs the excavation selected chosen by the student. If the answer selected by the student is correct, the excavator prepares a perfect excavation. If, on the other hand, an incorrect excavation is selected, the excavator will dig a few feet and then fall over to simulate a caving effect that, in many conditions, would likely result from incorrect excavation. Creating an Excavation Training simulator in a CVRE provides students with the ability to make choices, see the results of those choices, and hopefully, with repeated use, become proficient in excavating which would cost a construction company.

After more information on how an excavation should be done and the requirements for implementing the simulation was obtained, the next step was to determine an efficient design for the simulation.

Preliminary Design

The case study was designed using the knowledge of another research project, which allowed a user to drive an excavator in the CVRE by altering control buttons in LabVIEW, a simulation software. In order to do this, the CVRE polled LabVIEW every one second to determine if there was a request to make a change in the CVRE, calculate the new change, send the instructions on how to change to the CVRE, and display the results in the CVRE. The communication between LabVIEW and the CVRE was a two way communication made possible through a RESTful Webservice. The diagram for this particular project as seen in Figure 4 below would help provide a basis for how to approach the case study for this dissertation.

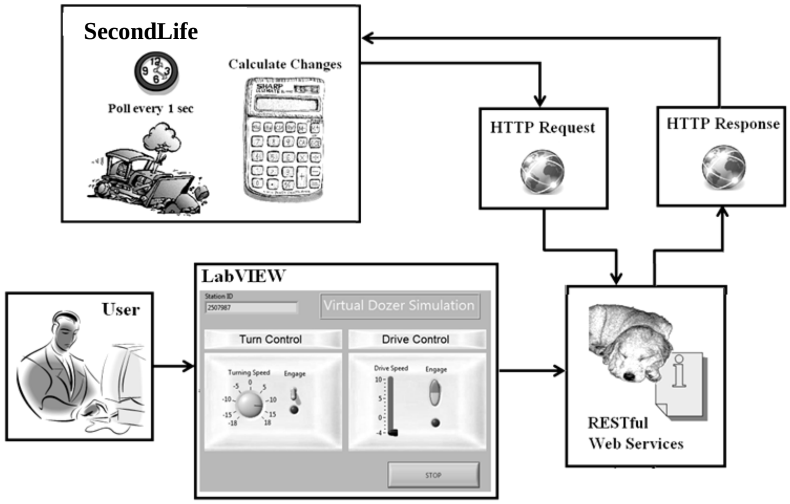


Figure 4. Design of the excavator driven by the user in LabVIEW.

To design the excavation training simulator, the next step was to design the communication protocol using the driving excavator project design. The changes to be made to the existing design included the following:

1. Transferring the calculation from the CVRE into LabVIEW;
2. Developing a communication protocol to interpret incoming messages and forwarding the messages to the correct receiver;
3. Developing a state machine in LabVIEW to determine what action should be calculated based on the message received; and
4. Developing a response adapter in the CVRE so that users can trigger a state.

The changes made to the excavator driving project can be seen in Figure 5 below.

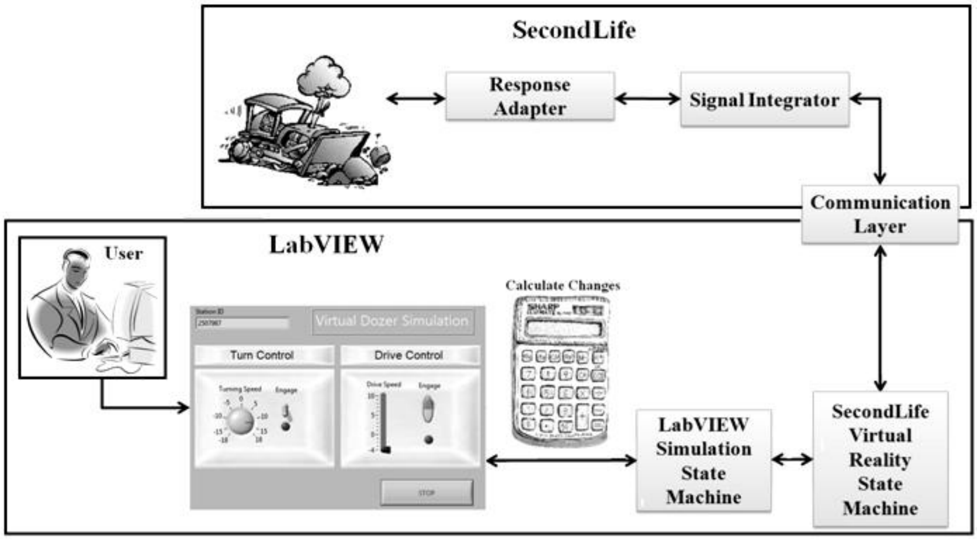


Figure 5. Design of the driving excavator project enhanced.

This exercise was helpful in determining the design of the excavation training simulator as it was a familiar project and all issues could be resolved by stepping through the exercise. Once the components for the communication protocol were determined using the driving excavator project, the next step was applying this knowledge to the case study.

When designing the case study, more finite details were added to the design to make it easier to implement. The process for the case study is as follows:

1. User is presented with the bid project information;
2. User pushes a button in the CVRE which triggers a state;
3. A message indicating what button was pressed is sent to IronPython;
4. IronPython determines what needs to be sent to LabVIEW and sends the message;
5. LabVIEW receives the message, calculates the response and sends this response to IronPython;
6. IronPython determines and sends the data required to update the environment to the CVRE; and

7. The CVRE is updated.

For example, if the user chose to press the excavate button in the CVRE, a message is sent to IronPython indicating that the user wants to excavate the terrain based on their selection. IronPython then sends a message to LabVIEW asking how to excavate, LabVIEW calculates how the excavation should be done, sends the results to IronPython, IronPython forwards the message to the CVRE, and the CVRE begins excavation. The design of the case study using the method can be seen in Figure 6 and is discussed in depth in the next section.

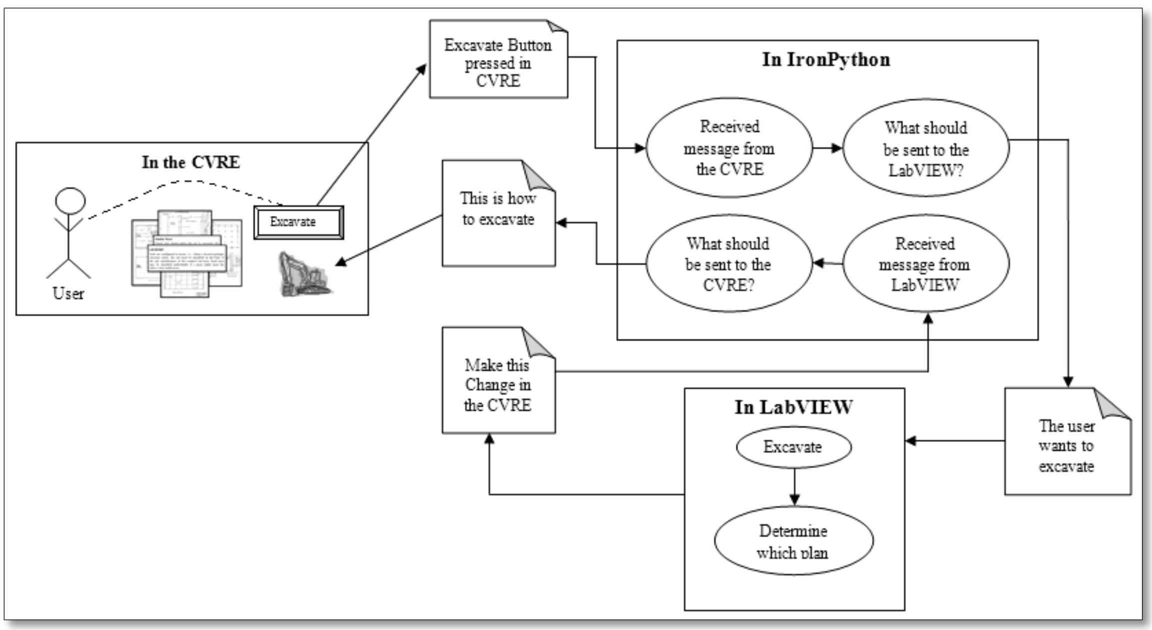


Figure 6. Design of the excavation training simulator.

Code Implementation

To implement the excavation training simulator, the following tasks were developed:

1. Determined and installed the software needed;
2. Created a construction environment in the CVRE;

3. Implemented the simulation through code;
4. Developed the animations within the CVRE;
5. Client and Server Deployment; and
6. Testing Against Several Performance, Development and Usability Metrics.

Determination and Installation of the Software Needed

In order to implement the simulator in the CVRE, several programs and programming languages were required. The necessary programs included a collaborative virtual world platform, and a viewer for logging into and interacting with in the CVRE. After looking into several CVREs, realXtend an open source project containing extended features of two other widely used CVREs, SecondLife and OpenSim, realXtend was selected because it was included with a generic virtual world server, an established region, examples of code, user documentary and an environment viewer. For an overview of the realXtend architecture the Introduction to realXtend Server Platform page can be accessed via the realXtend wiki (realXtend, 2010).

The three programming languages required to implement the simulation for realXtend, included LabVIEW, IronPython, and Linden Scripting Language (LSL). While LabVIEW was installed following general installation guidelines, IronPython and LSL was included in the installation of the CVRE server. Installing the CVRE server required installing the empty stand alone server provided by realXtend, setting up the stand-alone server, installing the CVRE viewer provided by realXtend, creating accounts for a test user, and testing the server setup.

In order to install the empty, standalone server, the base realXtend Server was downloaded from realXtend.org. The file downloaded was a .zip file which contained all files required to develop a realXtend CVRE. Hosting the server on the local machine required determining the static IP address of the machine and altering the configuration files in the realXtend server folder so that the virtual environment could be executed external to the local machine. Three important files in the server folder include the authentication server, avatar storage server, and the realXtend server. The purpose of the authentication server is to keep records of all users and handle authentication. This allows for new users to be added or deleted and provides a means for keeping track of users who enter the environment. The avatar storage server stores and delivers avatar data. The purpose of this file is to keep track of the appearance of the avatar, where it travels while in the CVRE, and other avatar maintenance and update utilities. The realXtend server handles in-world simulation as it shows all information happening within the environment such as objects being added, deleted, terrain being edited, etc.

The only way to determine if the server was set up correctly was to download the realXtend viewer, create accounts, and try to log into the CVRE. The viewer was downloaded from the realXtend.org website and installed following the step by step installation guide. Before anyone could enter the CVRE, an account had to be made, which required typing “create user” in the Authentication command window. Entering this command resulted in being prompted to fill in user information as seen in Figure 7.

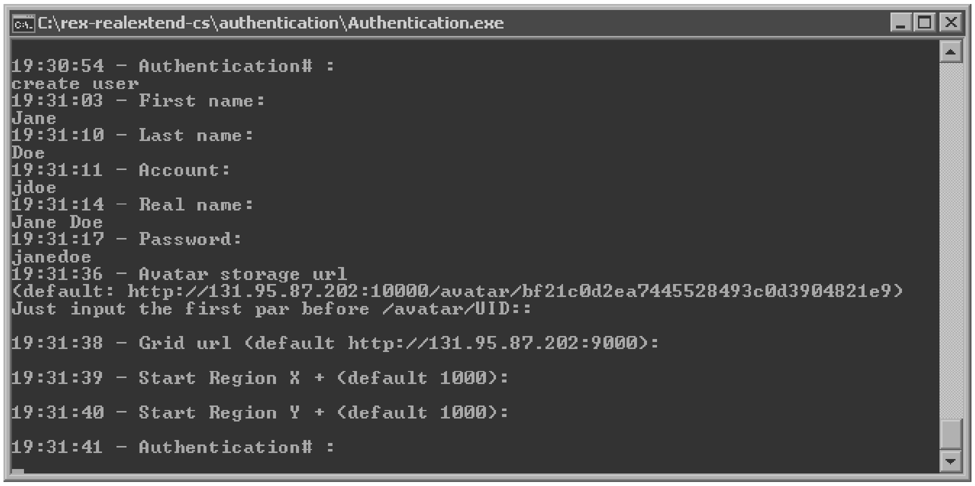


Figure 7. Creating a user in the authentication command prompt.

Once a user was created, a successful attempt was made to log into the CVRE. Logging into the CVRE was an indication that the server was set up correctly. Once the CVRE server was set up, the environment needed to be edited so that it would resemble that of an excavation site.

Creation of a Construction Environment in the CVRE

The base environment in realXtend is very bland because it is a 256 x 256 meter area, known in realXtend as a region, with a small amount of land bordered by water as seen in Figure 8.

Figure 8. Initial view of the terrain.

In order to give the CVRE the appearance of a real-world excavation site, objects like those found surrounding an excavation site needed to be added to the environment.

The desired layout for the excavation project consisted of dividing the region in half and using one half to represent a developed city and the other half to represent the construction area. It was evident that more terrain would be needed than was initially provided, so the "terrain fill x" command was used, which set the height of the terrain to x meters. Because the terrain would eventually be edited by an excavator (causing indentions in the terrain), the height chosen was 26 meters and thus the command entered in the Open Sim Launch executable was terrain fill 26.

After creating plenty of design space, by filling the 2256 x 256 region of land with terrain, actual models had to be imported into the region. Importing models into the region was a multi-phase process. First, the desired 3D model was imported from Google Sketchup, and converted into a mesh file suitable for importing into the realXtend environment. During this process, extra materials were extracted from the model and the model was sized to an optimal height for importing it into the CVRE. Next, the model was imported into the environment in two parts: the actual model, which was a mesh file, and the materials associated with the 3D model. Once all files were imported, a prim, the simplest object in the environment, was created and made to take the shape and color(s) of the desired 3D model using a built-in tool. After the desired model was created, it was sized to the desired shape and positioned in the desired location. To make the model more realistic, it was given the ability to cast shadows and reflect shadows by modifying the materials using existing tools within the CVRE. The finished project of the excavation site can be seen in Figure 9.

Figure 9. Images of the excavation site in the CVRE.

After the environment was decorated to resemble that of an excavation site, the next task to develop the excavation training simulator for the CVRE.

Implementing the Simulation Through Code

Implementing the simulation in the CVRE required the use of several subtasks. These subtasks included implementing the simulation in LabVIEW and implementing a communication protocol between LabVIEW and the CVRE.

Implementing the simulation in LabVIEW. In order to implement the simulation in LabVIEW, it was necessary to develop a database for storing project bid plans and intermediate results, implement a means for displaying the user with the bid project information, a site of bid plans and a soil type, as well as a method for allowing users to submit answers to questions via push buttons resulting in feedback. This design led to the idea of letting LabVIEW act as a state machine in which the actions come from the CVRE, the reaction is determined in LabVIEW and results are sent to the CVRE.

The database was developed using Microsoft Access for creating the tables and LabVIEW's database connectivity toolkit for connecting to the database from within LabVIEW. The database had several tables including one to store the combinations of bid plans and soil types, one for storing the answer to the current question, one for storing the users' answer to each question, etc. Figure 10 below shows an example of the "BidPlansAndSoilTypes" table in the database that stores a number to represent each set of bid project information, five views of each different project bid plan, a soil type, and the correct answers to each bid project plan.

Number	View1	View2	View3	View4	View5	SoilVie	Soil	Slope	Volume
0	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	GeologicalStrata	1.5:1	123
1	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	StableRock	0:1	124
2	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	TypeA	0.75:1	125
3	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	TypeB	1:1	126
4	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	TypeC	1.5:1	127
5	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	GeologicalStrata	1.5:1	128
6	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	StableRock	0:1	129
7	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	TypeA	0.75:1	130
8	C:\Users\strel	C:\Users\st	C:\Users\stre	C:\Users\s	C:\Users\	C:\Users\	TypeB	1:1	131

Figure 10. A portion of the "GetBidProejctInformation" table in the database.

The first step in implementing the simulation involved presenting the user with the bid project information, which required implementing a randomized generator to select a random set of bid plans and a random soil type from the database. This event would prevent multiple users from working on the same problem or a single user from frequently practicing the same example.

After determining how to present the bid project information to the user, the first question, one that required the user to determine the soil type described, was implemented in code. In order to do this, buttons were needed to represent each of the three possible choices, Type A, Type B, and Type C. Code was implemented so that when a user selected the soil type by pressing the corresponding button, the choice selected was compared against the soil type presented and, if correct, an indicator in the form of a round light, turned green. Otherwise, the indicator remained off until the correct choice was selected.

The second question implemented in the exercise asked the user to select the appropriate slope method. In the excavation process, several slopes can be chosen, but only certain slopes can be used with certain soil types. For example, two sloping methods are a simple slope and a vertical sided lower portion. While the simple slope can be used for excavations with Type A soil, a vertical sided lower portion slope cannot. This condition required two cases for checking the user's answer. For the first case, assume that the user identifies the soil type correctly. This is the simplest case, as the slope method is checked against the soil type presented and the selection made by the student in order to determine if the slope method is possible for the given soil type. In the second case, assume that the user did not identify the soil type correctly; this is the more complicated case as it has to be decided to either check against the soil type presented to the user or check against the soil type selected by the user. For this implementation, the design specs required checking against the user's selection and indicating the answer was correct via another green indicator if the slope method was possible for the user selected soil type.

The third question implemented into the example/program asked the user to determine the appropriate slope ratio based on the given soil type and selected slope method. Four buttons were created to represent each of the possible slope ratios. Verifying the soil ratio selected involved determining whether or not that particular slope ratio was appropriate for the slope method according to the soil type. For example, if given a soil of Type A, the simple slope can be used, but it must have a slope ratio of ¾:1 according to OSHA standards.

The last question implemented asked the user to determine the volume of soil to be excavated. Each project bid plan has a specific volume of excavation which is calculated based on the soil type and slope method selected; however, the user had to be presented the correct answer in addition to three other random answers. In addition, the correct answer could not be represented by the same button each time. Thus, a random generator was developed to determine the location of the three incorrect answers and which button would be associated with the correct answer. An image of the code and the executable can be seen in Figure 11.

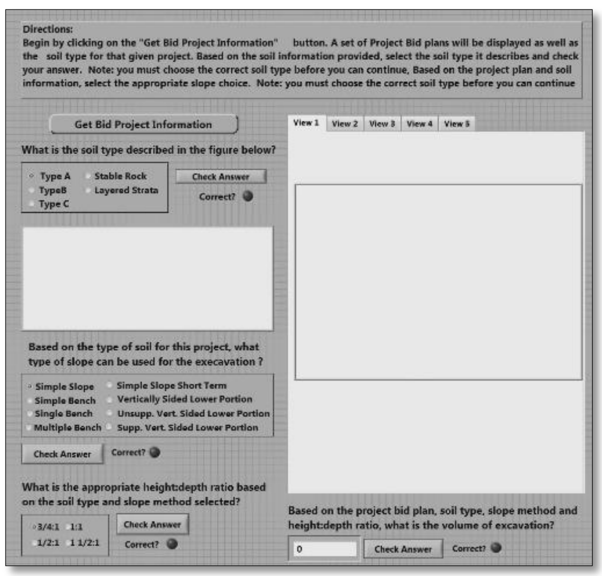


Figure 11. Front panel of the LabVIEW code.

The figure above shows the front panel view of the simulator, which allows users to interact with the program by making selections through pressing buttons and seeing the results of their answers. Developing code in LabVIEW is done using the back panel, which is not seen by the end user. Figure 12 shows an example of the code that determines a random bid plan and soil type from the database to present to the user by querying the database.

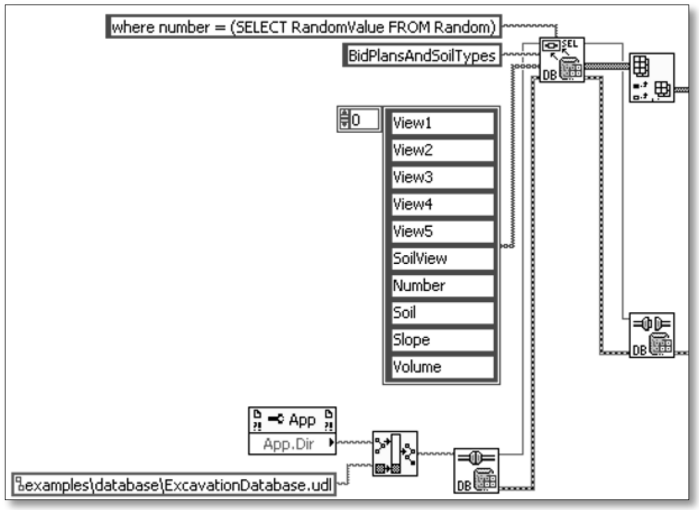


Figure 12. Sample of LabVIEW code for querying the database.

After implementing the code in LabVIEW, the next step was to determine how to send messages between LabVIEW and the CVRE. This required investigating the following:

1. TCP/IP communication tools in LabVIEW; and
2. The socket communication for realXtend.

On the LabVIEW side, an existing TCP/IP connection tool was used to connect to a specified IP address and wait for messages of a specified bytes site that needed to be sent or received. IronPython, RealXtends means for handling TCP/IP communications, had to be learned so that messages could be sent to the realXtend environment. Figures 13 - 16 show LabVIEWs means for connecting to a specified port, receiving a message, sending a messages, and terminating the connection. Figure 13 shows the TCP connection tool which specifies the port at which the socket should connect and an initial null error log.

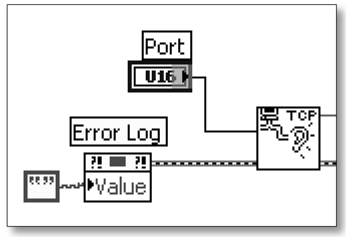


Figure 13. Sample of LabVIEW connection tool.

The TCP connection tool contains all the necessary components required to connect to the specified socket implemented in various LabVIEW functions. The next sample of LabVIEW code shows how messages are received and processed as seen in Figure 14.

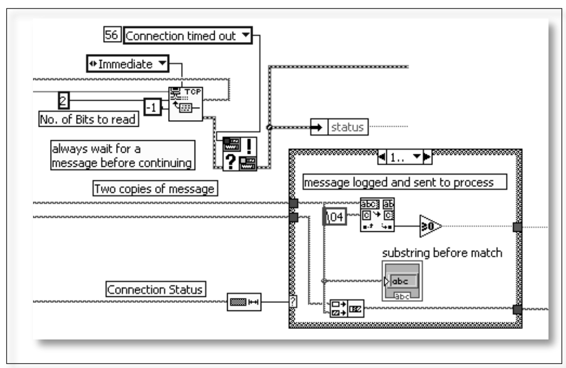


Figure 14. Sample of LabVIEW code for receiving message.

For the purpose of the case study, the number of bits to be read is indicated by the 2 and the amount of time to wait on a message was infinite, indicated by the -1, as a message was the driver of the necessary actions. Once the message was received, it was added to a log that kept up with all incoming messages and forwarded on to other sections of the code responsible for determining how to handle the incoming message. The next sample of code as seen in Figure 15 shows the tool used to send messages to a specified port.

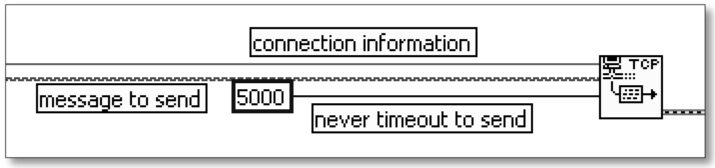


Figure 15. Sample of LabVIEW code for sending a message.

The communication channels going into the TCP connection tool are the socket information, the actual message, and the time to wait before sending a message. The next sample of code as seen in Figure 16 shows the TCP connection being terminated.

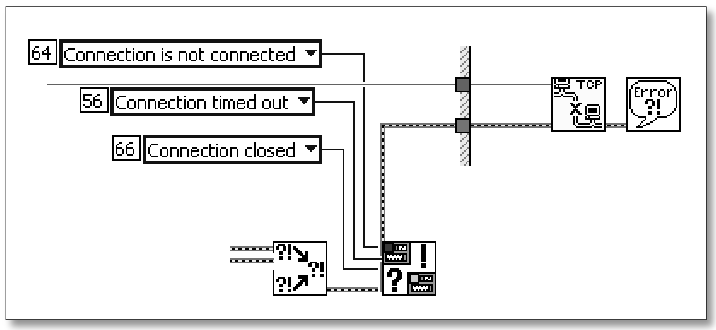


Figure 16. Sample of LabVIEW code for terminating connection.

To promote safe programming practices, LabVIEW provided the means to terminate the TCP connection gracefully if the connection never was connected, the connection timed out, or the connection was closed.

Implementing the communication protocol. To handle the communication between LabVIEW and realXtend, IronPython code had to be developed. Below is a segment of code written in IronPython that connects to a specified socket, receives data, and sends data to the intended receiver. The first sample of code as seen in Figure 17 is used to handle the creation of prims in the CVRE. When prims are connected, they must form a connection to a socket so that they can begin listening for commands.

```
#A connection needs to be made when an object in the CVRE is created
def EventCreated(self):
    global f

    try:
        s.connect(("127.0.0.1",6342))
    except socket.error,msg:
        print "Already connected

    f = s.makefile('w',1)
    socketOpen = 1
    super(self.__class__,self).EventCreated()
```

Figure 17. IronPython code to handle EventCreated in RealXtend.

The next set of code as seen in Figure 18 destroys the connection to a socket if the prim is deleted in the CVRE.

```
#What needs to be done in realXtend when we delete an object that is connected to an object
def EventDestroyed(self):
    print "LabviewTools.Connection EventDestroyed"
    super(self.__class__,self).EventDestroyed()
```

Figure 18. IronPython code to handle EventDestroyed in RealXtend.

The next segment of code as seen in Figure 19 is used to handle what happens when prims in the environment that need to communicate with LabVIEW are touched.

```
def EventTouch(self,vAvatar):
    global f
    #write to realXtend command line to know that a button is pressed
    print "Button Pressed"

    #get the id of the object that wants to communication
    myid = str(self.llGetObjectName())

    # If the message is Excavate, then a 26 is sent to LabVIEW to signify the CVRE needs
    #instructions on how to excavate (s socket) and the when the instructions are received
    #back from LabVIEW, they need to be sent to the CVRE.
    if(myid == "Excavate"):
        s.send("26")
        f.flush()
        message = s.recv(1024)
        self.llShout(0, message)
```

Figure 19. IronPython code to handle EventTouch in RealXtend.

When a prim that needs to communicate with LabVIEW is touched, its name is determined and based on its name a specific instruction is sent to LabVIEW and sometimes a message is returned back to the IronPython code to forward to the CVRE. For example, if the user presses the excavate button in the CVRE, they are really pressing an invisible prim named Excavate as seen in Figure 20.

Figure 20. Excavate button on the LabVIEW front panel with an invisible prim called overlay

Pressing this button in the CVRE sends the message Excavate to IronPython which triggers a 26 to be sent to LabVIEW on the socket and IronPython waits on a message to come back from that socket so that LabVIEW can tell the CVRE what needs to happen.

After implementing the code required to present the user with an interactive environment and the communication for sending messages between LabVIEW and the CVRE, the next step was to determine how to implement the simulation section of this case study, an excavation of the environment based on the students' answers. This required learning more about the LSL programming language built inside the realXtend CVRE. The generalized task was to determine what existing LSL functions could be used on a prim so that the excavator would move along a specified path, altering the terrain height on that path.

To implement the excavator in the CVRE, functions were used that would read the plan to be excavated from the body of an html page containing the number of rows, and columns to be excavated and the depth to be dug at each point. These html pages were created by exporting an excavation plans from REVIT. From this information, one prim representing an invisible excavator dug the environment and another prim holding the excavator model moved in the path to appear as if it was actually doing the excavating. The two prims, one for the excavator and one for the digger, can be seen in Figure 21 where each prim is highlighted with white boxes.

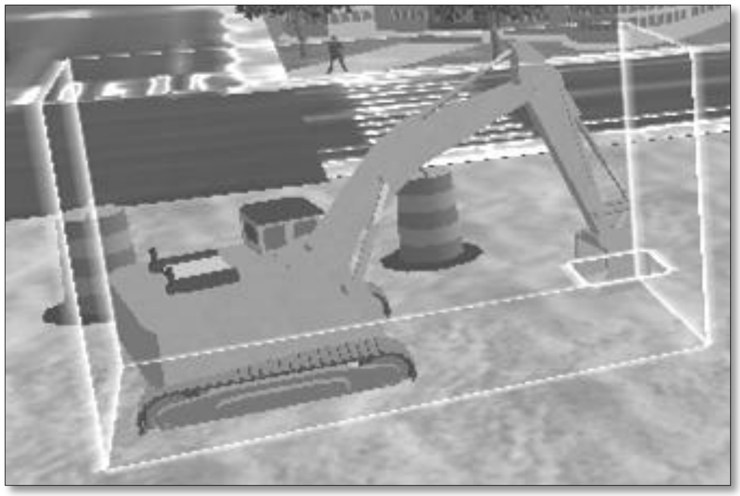


Figure 21. One prim for excavating and one prim for holding the excavator model.

The invisible prim was needed as the excavator had to move up and down to change the location at different heights which would have given the excavator the unnatural appearance of jittering. Below are samples of code that allow the excavation to be prepared through using the invisible prim. The first sample of code as seen in Figure 22 is the state entry code which sets up two listeners, one to listen for a URL containing the excavation text and one for communicating with the invisible prim.

```
state_entry()
{
    llSetText("", <1.0, 0.0, 0.0>, 4);
    llListen(0, "", NULL_KEY, "");
    llListen(242,"",NULL_KEY,"");
}
```

Figure 22. Invisible excavator prim code to initialize listeners.

The second sample of code as seen in Figure 23 handles messages coming in on the different channels.

```
listen(integer channel, string name, key id, string message)
{

        if(channel == 0)
        {
            http_request_id = llHTTPRequest(url, [], "");
            llSay(0, "File has been read. Say 'begin dig' when ready...");
            original_pos = llGetPos();
            llSetTimerEvent(0.001);

            digging = 1;
            mapPos = 1;
            colCount = 1;
            rowCount = 1;
            url = message;
            rposition = hposition;


            llSay(0, "File name received: " + message);
            llSay(0, "Say 'name correct' if this is correct...");
        }

        if(channel == 242)
        {
            vector diggerhome = <108.008, 134.913, 28.123>;
            if(message == "LetsRelocate")
            {
                while ( llVecDist( llGetPos(), diggerhome ) > 0.001)
                {
                    llSetPos( diggerhome );
                }
            }
        }
}
```

Figure 23. LSL Code to listen for messages on different channels.

The message that comes in on channel zero is a URL that is stored in a variable so that it can be used by other functions. This URL stores the instructions on how to dig the excavation containing the number of columns, rows, and the depth to dig at each point. On the other hand, if the channel communicating is channel 242, this is the excavator prim telling the excavator that the excavation is complete and they need to relocate back to home. The third sample of code as seen in Figure 24 is responsible for reading the entire HTML body using a special LSL command and storing the number of rows and columns to be excavated.

```
http_response(key request_id, integer status, list metadata, string body)
{
    if (request_id == http_request_id)
    {
        llSay(0, "\n" + body);
        mapText = llParseString2List(body,[" ", "\n"],[]);
        rows = llList2Integer(mapText, 0);

        columns = llList2Integer(mapText, 1);
        rowCount = 1;

    }
}
```

Figure 24. LSL Code to read HTML content.

The fourth sample of code as seen in Figure 25 is the code responsible for actually excavating the site.

```
timer()
{
    rowNum = rowCount + 1;

    while(rowCount != rowNum & rowCount <= rows)
    {
        ++mapPos;
        position = llGetPos();
        depth = llList2Float(mapText, mapPos);
        llSetPos(<position.x, position.y, original_pos.z + depth>);
        llModifyLand(LAND_LEVEL, LAND_SMALL_BRUSH);
        llSetPos(position + <1.0, 0.0, 0.0>);
        llSay(22, (string)"increase_x");
        ++colCount;

        if(colCount == columns + 1)
        {
            colCount = 1;

                rposition = rposition + <0.0,1.0,0.0>;
                llSay(22,"moveover");
                while ( llVecDist( llGetPos(), rposition ) > 0.001)
                {
                    llSetPos( rposition );
                }

            ++rowCount;

            if(rowCount > rows)
            {
                llSetTimerEvent(0.0);
                 digging = 0;
            }
        }
    }

    if(rows == 3 & rowCount == 4)
        llSay(22,"flipoverrover");
```

Figure 25. Invisible excavator prim code that excavates the terrain.

To dig the terrain, the data values in the HTML body are read and the land altered to that particular height using the llModify command. In addition to altering the terrain, this code is responsible for telling the excavator how to move in sync with the invisible excavator. The code within the excavator prim is very similar to the invisible prim as seen in the following three figures. As seen in Figure 26, the excavator prim sets up a listener on channel 22 for the purposes of receiving instructions from the invisible excavator prim on how to move in order to excavate.

```
state_entry()
{
    diggerrelocate = diggerstart;
    llListen(22,"",NULL_KEY,"");
}
```

Figure 26. Excavator prim code to initialize the digger.

The second sample of excavator prim code as seen in Figure 27 processes the messages coming from the invisible excavator prim which either want the excavator to move in the x or y direction or flip over to indicate a bad excavation choice.

```
listen(integer channel, string name, key id, string message)
{
    if(message == "increase_x")
        llSetPos(llGetPos()+<1.2,0.0,0.0>);

    if(message == "moveover")
    {
         diggerrelocate = diggerrelocate + <0.0,0.35,0.0>;
         while ( llVecDist( llGetPos(), diggerrelocate ) > 0.001)
         {
             llSetPos( diggerrelocate );
        }
    }

    if(message == "flipoverrover")
    {
        vector r = <-45.0,0.0,180.0>;
        r *= DEG_TO_RAD;
        rotation quat = llEuler2Rot(r);
        llSetRot(quat);
    }
}
```

Figure 27. Excavator prim code to handle messages sent by the invisible excavator prim.

The last sample of excavator prim code as seen in Figure 28 was developed to help the excavator prim and the invisible excavator prim return to a home position after an excavation.

```
touch_start(integer x)
{
    vector startPosition = <113.862, 137.281, 31.361>;

    llSay(242,"LetsRelocate");
    vector p = <0.0,0.0,180.0>;
    p *= DEG_TO_RAD;
    rotation quat = llEuler2Rot(p);
    llSetRot(quat);

    while ( llVecDist( llGetPos(), startPosition ) > 0.001)
    {
        llSetPos( startPosition );
    }
}
```

Figure 28. Excavator prim code to return the excavator prim and invisible excavator back to the start position at the end of an excavation.

Another part of the simulation process required developing an excavator for flattening the terrain in between excavations. This was done similar to the digging excavator as one prim was for the flattening and another for digging. The difference in the excavator code is that more than two prims were needed to flatten the terrain and the code was easier to implement. Figure 29 shows the excavator and five of the numerous land leveler prims, the same invisible prims required to flatten the terrain.

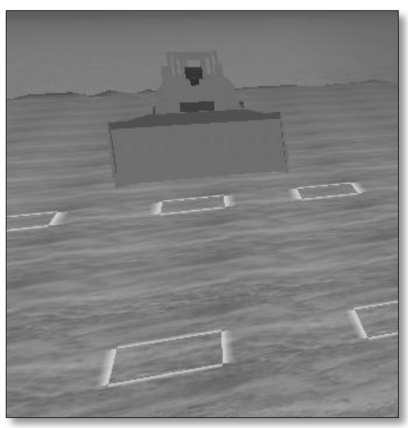


Figure 29. Excavator prim and invisible prims for leveling the terrain.

Below are some samples of code required to flatten the terrain using these prims. The first sample of code is the code inside the land leveler which listens on channel three for a signal to begin flattening the terrain. This command occurs when a user presses a button in the environment. When this message is "heard," a different message is sent to all land leveler prims to tell them it is time to flatten the terrain. The code in the land leveler prims as seen in Figure 30 are nothing more than instructions to listen on a channel for the message to begin leveling and the leveling the terrain using the same command from the excavator prim, llModify().

```
default
{
    state_entry()
    {
        llListen(3, "", NULL_KEY, "");
    }

    listen(integer channel, string name, key id, string message)
    {

        if (message == "flatten")
        {
            for(j = 0; j<maxc;j++)
            {   llSetTimerEvent(600.0);
                llSetPos(llGetPos() - <1.0, 0.0, 0.0>);
            }
            llSetPos(llGetPos() + <10.0, 0.0, 0.0>);
            llSetPos(llGetPos() + <10.0, 0.0, 0.0>);
            llSetPos(llGetPos() + <10.0, 0.0, 0.0>);
            llSetPos(llGetPos() + <10.0, 0.0, 0.0>);
        }
    }
}
```

Figure 30. Land leveler prim and invisible prims for leveling the terrain.

Implementing the excavation simulator took approximately nine months. The majority of this time was spent on the following:

1. Creating the construction site in the developing the code in LabVIEW; and
2. Developing the simulation within the CVRE.

Developing the code in LabVIEW took approximately four months. The reason LabVIEW took this amount of time is because LabVIEW did not always have a straight forward solution for implementing the problem in code and thus a work around had to be developed. For example, when LabVIEW code is executed, all code on the panel is executed, this results in variables being overwritten. While LabVIEW does provide a solution for this by providing a tool that will execute segments of code in a sequence, there is not a means for passing these variables between two sequences that exist in different if cases. To solve this issue, the database was used to store intermediate values. Developing the simulation with the CVRE took approximately two months. During this time period, the majority of effort went into determining how to read the data from the web page, which LSL functions were needed to develop the simulation, and how to tell objects what needed to be done. After looking at several LSL code samples that read from the html body, a method for reading the information from the website was implemented. Determining what LSL functions to use was more involved as some functions have limited capabilities. For example, when using the locate function, an object can only be moved 10m per each call of the function, which required developing an algorithm to determine how many times to call this function for a specified amount. Determining how to tell objects within the CVRE required studying how messages were passed between objects within the environment. To send messages between two different objects required ensuring that a message intended for a specific object was broadcasted on a certain channel and that the object sending as well as the object are close enough to hear the message. Setting up this communication was very time consuming as numerous channels to communicate were required in addition to many objects that needed to communicate. This required determining how to broadcast messages to the indicated prim without interacting with other prims.

The above are just a few of the tasks where most of the time was spent developing the excavation training simulator in the CVRE. Table 4 provides the timeline used for this case study.

Table 4

Case Study Timeline

Task	Time
Designing the simulation. Included meeting with Dr. Rapp to finalize the design, becoming familiar with the excavation process, and developing the flowchart for the simulation design.	1 month
Creating a construction environment in the CVRE. This included importing the models from sketchup, assigning each model to a prim, assigning shadows to materials and materials to its prim, sizing prims and placing prims in the appropriate location.	1 month
Implementing the simulation. This included developing the database for storing information and the LabVIEW code	4months
Developing the animations within the CVRE. This included developing the code to flatten the terrain, excavate the terrain, and setting up communication between multiple prims.	2 months
Finalizing, Replicating, and Testing the case study. This includes making last minute changes to the design, developing the surveys, replicating the developed code onto servers, and the testing day	1 month

Client and Server Deployment

After the case study was implemented, the next step involved migrating the code onto a server so that the excavation training simulator could be tested by Dr. Rapp's students. Migrating the code to the server required performing a physical to virtual (P2V) conversion which involves creating a virtual hard disk (VHD) of the desired volume. A VHD contains what is found on a physical hard disk drive, such as files, folders, file system, and disk partitions. In order to make the VHD, Disk2Vhd which uses Windows' Volume Snapshot capability to create consistent point-in-time snapshots of the volume, the complete C drive of the machine used to implement the code, to be included in a conversion was used.

After making the VHD, the volume was optimized by removing unused files and running CCleaner which is an open source tool that would remove all unused files from the volume that might have been overlooked during the initial unused file removal. Removing all unused files required determining which software and files were needed to run the case study and removing all files that were not part of this list. On the other hand, the CCleaner tool removes not only the unused files but also traces of online activities and includes a cleaner for the registry. For example, the internet cleaner removed temporary files, history, cookies, download history, and from history where as the registry cleaner will removed unused and old entries including file extensions, ActiveX Controls, ClassIDs, ProgIDs, Uninstallers, shared DLLs, Fonts, Help Files, Application paths, Icons, Invalid shortcuts and more.

After optimizing the VHD, a hypervisor had to be set up on the server which allows multiple operating systems to run concurrently on a host computer. This process is commonly referred to as hardware virtualization and was used to help consolidate the server as well as enable a high-performance virtual environment. Implementing the hypervisor required insuring that the network connections were correctly configured and installing Microsoft's Hyper-V role.

Once the Hyper-V role was installed, the last step was to create virtual machines and install the guest operating system, the VHD, on the virtual machine. In order to create virtual machines, the Hyper-V manager was used, which is an easy step-by-step process for creating virtual machines. The last option in the Hyper-V manager asks for the "Installation options," which is where the location of the VHD is specified.

Setting up the virtual machines provided five CVREs which could be used to test the performance, development, and usability metrics. The addresses for each of these virtual machines include: 131.95.87.202, 131.95.87.203, 131.95.87.204, 131.95.87.205, and 131.95.87.206.

Testing Against Several Performance, Development and Usability Metrics

Before testing could begin, students were provided with a link to a website containing their access information and a set of instructions on how to run the experiment. In addition to the website, Dr. Rapp provided an explanation of what was to be encountered. In addition to making sure that students were aware of what was to be encountered, two surveys were prepared for the students to answer upon completing the exercise. Using LimeSurvey, two different surveys were developed to test the Usability of the software and the motivation.

Measuring the Usability

The statements that appeared on the usability survey are divided into seven categories: the suitability of the task, the self-descriptiveness, controllability, conformity with user expectations, error tolerance, suitability for individualization, and suitability for learning. To prevent the surveyed from detecting these seven categories, the questions were not grouped based on these categories. There were 24 statments that appeared on this survey where one statment from each category included:

- There were times when more information or additional tools were needed but unavailable.
- The instructions for each task were easy to understand.
- The instructional material was organized and easy to locate.

- The instructions and feedback were consistently arranged.
- It was easy to correct errors encountered in the simulation environment.
- The instructional material was available at convenient hours of the day.
- The instructional material provided opportunity for decision making.

Figures 31 - 37 display the results of the usability survey. Each table shows the number of students that completed the survey (n) and the percentage of each student that strongly disagreed, disagreed, undecided, agreed, or strongly agreed about the questions in each of the categories.

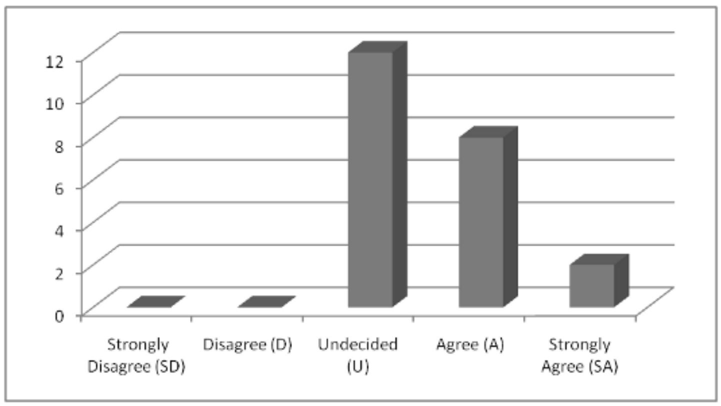


Figure 31. Chart displaying the extent of the suitability for the task.

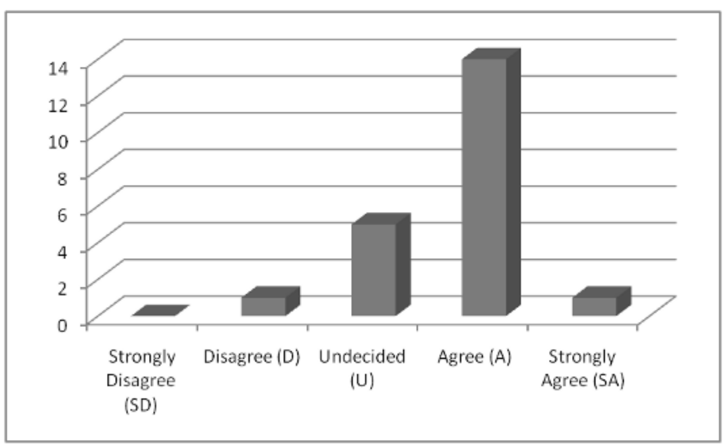


Figure 32. Chart displaying the self-descriptiveness of the activity.

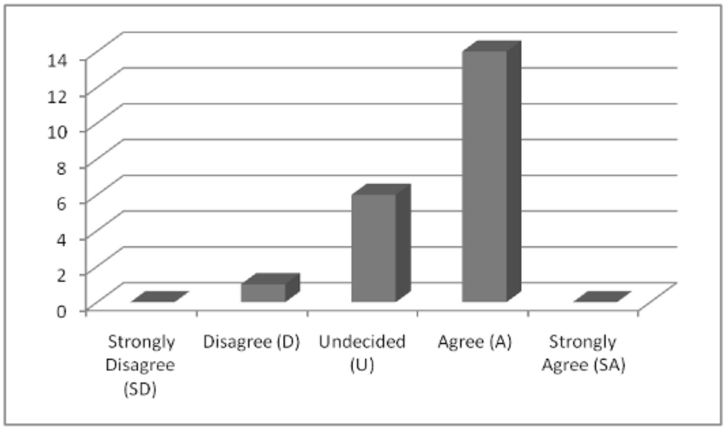


Figure 33. Chart displaying the extent of controllability felt by users.

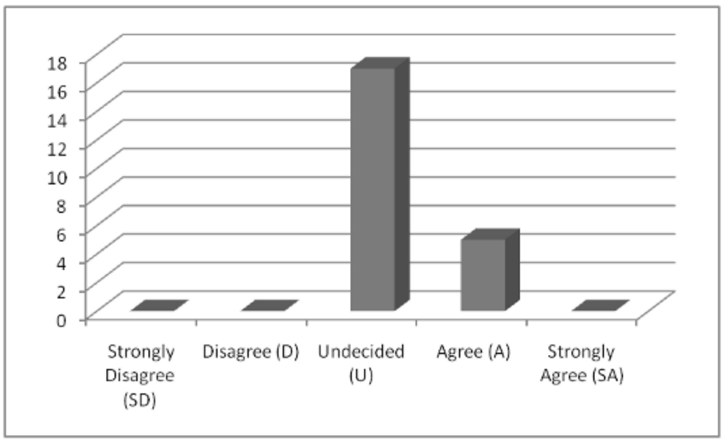


Figure 34. Chart displaying the conformability with the user's expectations.

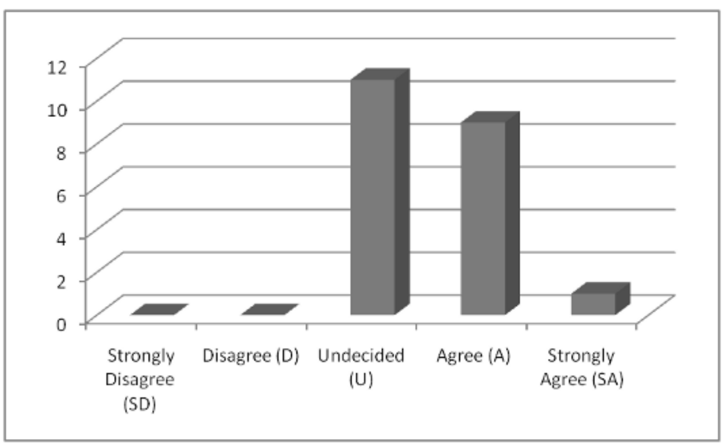


Figure 35. Chart displaying the error tolerance of the activity.

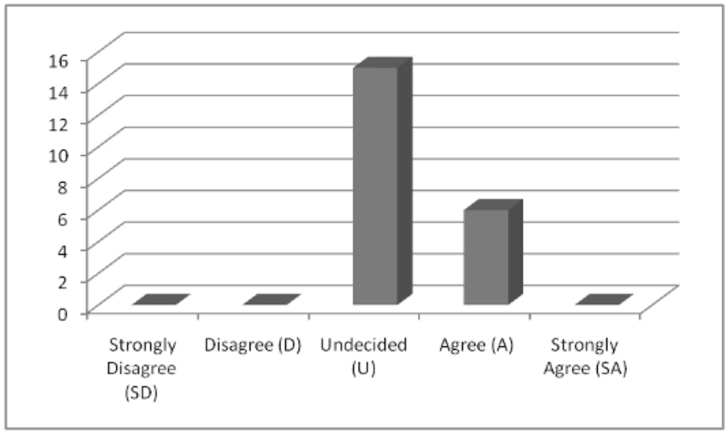


Figure 36. Chart displaying the suitability for the individualization of the activity.

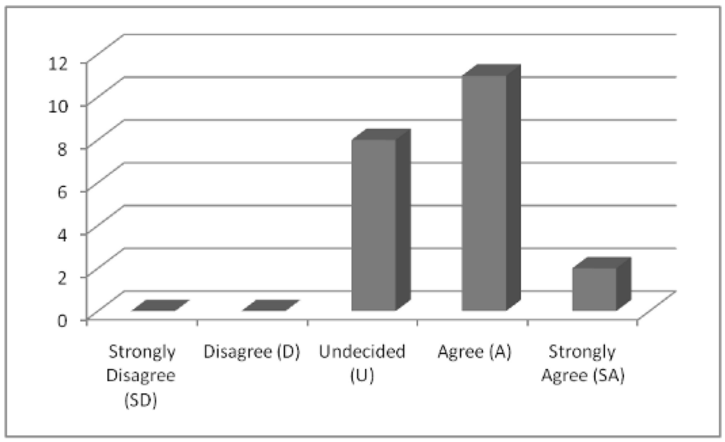


Figure 37. Chart displaying the suitability for learning for this activity.

Measuring the Motivation

The statments on the motivation survey were used to determine if the student was motivated in completing the activity. Like the usability survey, categories were used to help determine the extent of motivation. These categories include: Interest/Enjoyment, Perceived Competence, effort/importance, pressure/tension, perceived choice, value/usefulness. There were 36 statments that appeared on the motivation survey. Examples of statments on this survey from each of these categories included:

- This activity did not hold my attention at all.
- I couldn’t do this activity very well.

- I didn't try very hard to do this activity.
- I did not feel nervous at all while doing this activity.
- I didn't really have a choice about doing this task.
- I believe doing this activity was beneficial to me.

Figures 38 - 43 display the results of the motivation survey. Each table shows the number of students that completed the survey (n) and the percentage of each student that strongly disagreed, disagreed, undecided, agreed, or strongly agreed about the questions in each of the categories.

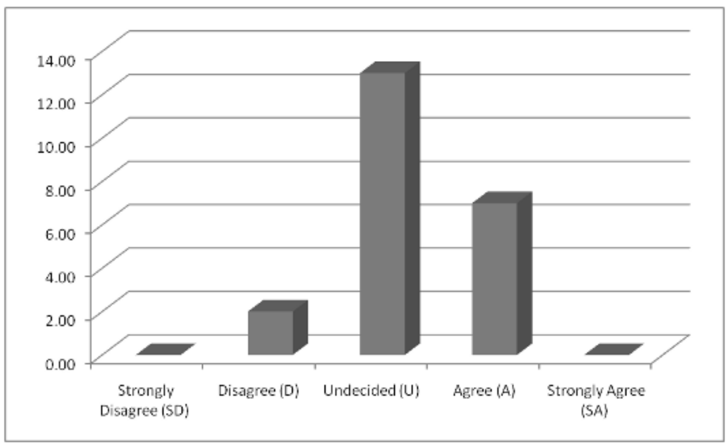


Figure 38. Chart displaying the interest and enjoyment of this activity.

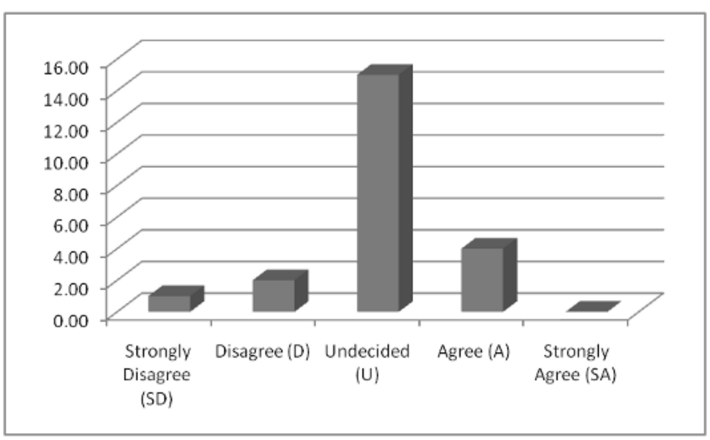


Figure 39. Chart displaying the perceived competence of this activity.

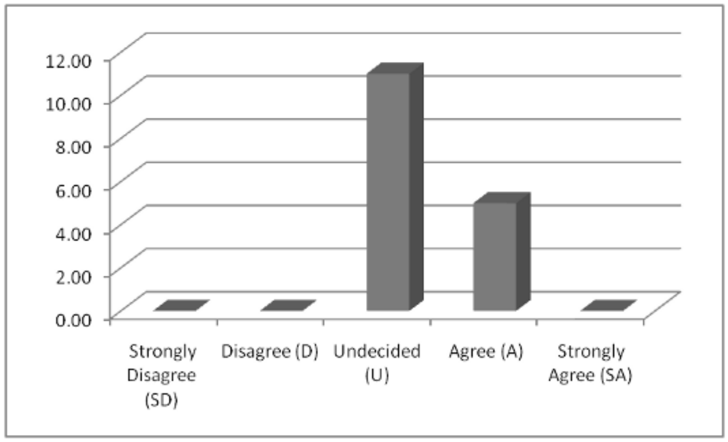


Figure 40. Chart displaying the effort/importance of this activity.

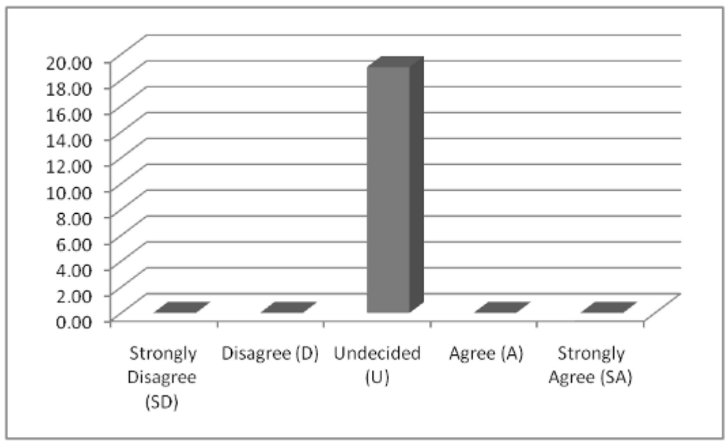


Figure 41. Chart displaying the pressure/tension of this activity.

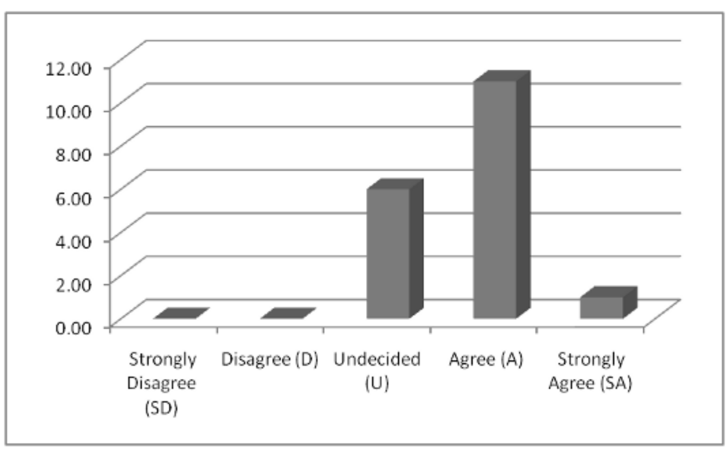


Figure 42. Chart displaying the perceived choice of this activity.

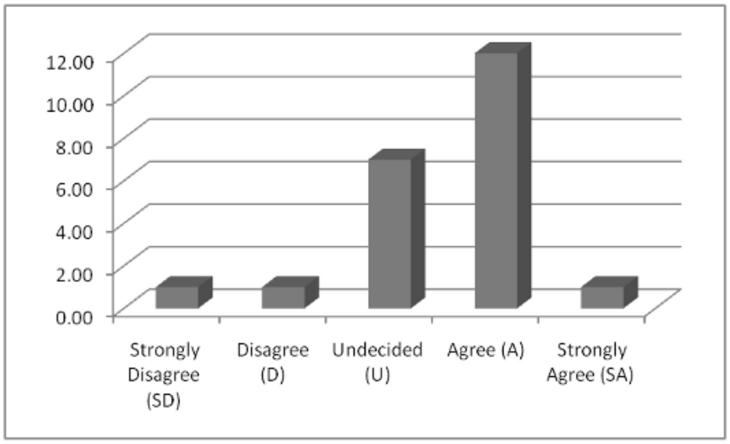


Figure 43. Chart displaying the usefulness of this activity.

In additions to the questions on this survey, students were also asked if they thought this activity should be available to upcoming students and should this activity be modified for the upcoming students. Also, if they decided that modifications were needed, they were asked to describe those modifications. In addition, the option to leave additional comments was included. While most students agreed that this activity should be available to upcoming students, all who were in agreement commented that the training simulator needed to be enhanced before it was made available to future students.

Measuring the Performance

In order to determine the performance of the excavation training simulator the OpenSim.32BitLaunch.log file was inspected. This file contains a list of all communication errors that occur within the environment. The only type of error that existed in this log file was an error relating to the improper initialization of the authentication server which did not have a relevance to this test case or affected the communication performance. An example of an error that occurred during this test can be seen in Figure 44.

```
ERROR -OpenSim.Region.CoreModules.Avatar.InstantMessage.PresenceModule
[PRESENCE]: Failed to notify message server of agent location for 285d985e-    bb39-
4a0f-    b344-b81519adf3d4
```

Figure 44. Example of communication error irrelevant to the simulation performance.

Analyzing the server log led to the conclusion that communication between the client and server was not interrupted. While an error existed in these logs, it was not relevant to the performance of the communication method. This error was where a user attempted to connect to the CVRE, which was expected as this was notified by the student in the comments section of the motivation survey. While the exact reason for this student not being able to connect is unknown, the primary cause of this situation is due to a bad data connection or a computer that does not have accurate hardware to support the CVRE graphics. In addition to measuring the communication success, the software was also stress tested as 22 students were allowed to login to the CVRE. All students were able to log into the CVRE without interrupting the communication method implemented.

Lessons Learned

From the information obtained through the surveys and performance tests, it was concluded that the excavation training simulator needed improvement in order to be an optimal training simulator for students. The changes that need to be made included:

- Ensure that all information within the environment is visible to the user.
- Make sure that the user obtains some intermediate format to ensure they feel user control.
- Implement a better method for ensuring that the user does not have access to any controls that could offset the results of the experiment.

Unlike the last two required changes, the first change does not reflect the design of the communication protocol. To make the communication protocol work appropriately, more restrictions need to be added to the code so that users can be aware of when they have interacted by the system and prevent simultaneous users at the same time from interacting with objects in the environment that should be inaccessible at some state in the simulation.

CHAPTER V

GENERALIZED COMMUNICATION PROTOCOL

As indicated in the methodology section of this dissertation, the purpose of the test case was to use a concrete case study to help derive the communication protocol between a collaborative virtual reality environment (CVRE) and a simulation language. After completing the test case, it was concluded that the structure of the communication protocol was a specialized version of client-server architecture. To design the communication protocol, the basic fundamentals of client-server architecture were studied so that the specialized components could be determined based on how the case study was implemented.

Defining the Client-Server Architecture

In the definition of client-server architecture, the three main components are a client, server and middleware. The purpose of the client is to send requests to the server and render incoming instructions whereas the purpose of the server is to wait on messages from the client to determine what information is needed by the client. In order for the communication to occur between the client and the server, a manager, the middleware, is required. It serves as the mediator between the client and server to ensure the receipt of messages. In order to maintain communication between the client and server in this type of architecture, the code implemented for each component requires certain elements.

The client code must have the ability to connect to a specified port that it will use for communicating with the server through the middleware. Through this port, it must be able to send messages to the server by way of the middleware and receive message from the server through a different communication port on which it listens on for incoming messages. Lastly, the client must be able to process the information requested from the server and act correspondingly. On the other hand, the server code needs the ability to listen to the port that communicates messages from the server by way of the middleware, determine the information requested by the client, and send this data back to the client through way of the middleware. The middleware code is perhaps the simplest piece of code as it only needs the ability to connect to the port for communicating with the client and server and pass the messages between them keeping in mind the sender and recipient. The basic layout of this generalized client-server communication can be seen in Figure 45.

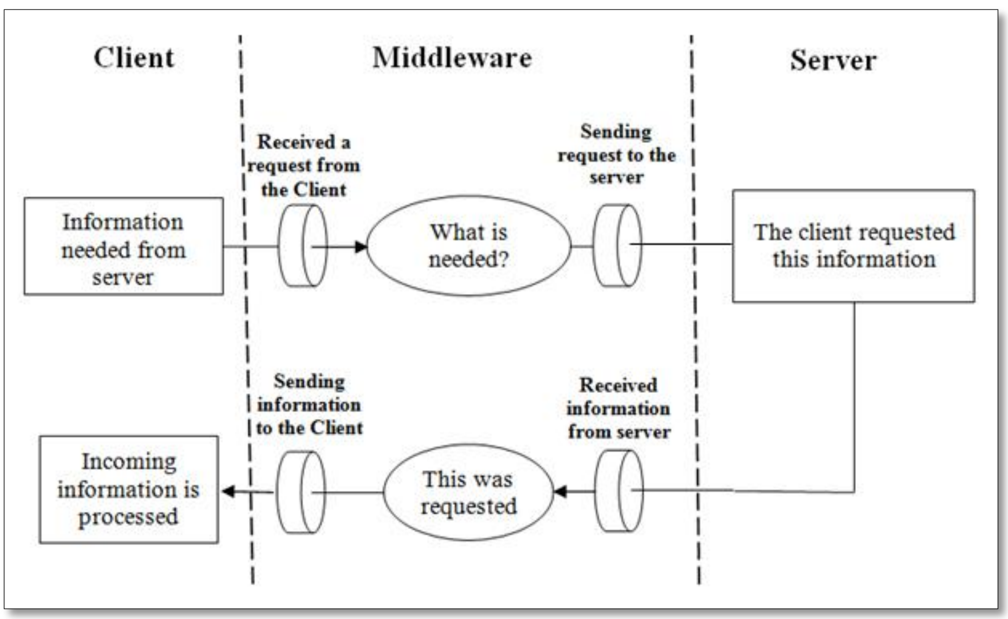


Figure 45. Diagram of the basic client-server architecture.

Specializing this basic architecture using the results from the test case, as discussed in Chapter IV, would lead to a hypothesized communication protocol.

The Specialized Communication Protocol

Developing a communication protocol using the existing client-server architecture required determining the three main components of the specialized client-server architecture that were similar to the generic client-server architecture. These three components to replace those in the existing client-server architecture were discovered to be the specialized version of the client which is the CVRE, the server which is the simulation language, and middleware which is any supported communication language between the CVRE and the simulation language. Each of the individual specialized components was scrutinized to determine the specific components so that a recipe for developing any simulation within a CVRE could be provided.

The specialized components for the CVRE were the event trigger(s), two event adapters, and two communication adapters. An event trigger in the CVRE was needed for indicating a selection, a transition condition, made in the CVRE. When an event is triggered a request is sent to the simulation to determine how the CVRE should react to this selection. In order to communicate with the simulation language a CVRE to Network communication adapter is required to pass the message from the CVRE to the simulation language. Once the reaction is determined, a message is sent through the middleware and received by the CVRE using the Network to CVRE communication adapter. The received reaction, known as an entry action, is processed by a primary event adapter in the CVRE which determines how to represent the visualization for the action calculated by the simulation language. In some cases, a secondary adapter may be required to assist the primary adapter as it will take on the action of a slave adapter in order to carry out commands requested by the primary adapter.

The specialized components required for the simulation language are a state machine and two communication adapters. The purpose of the “server” side state machine is to receive a state to be transitioned to from the CVRE, keep the current state of the simulation in the CVRE, interpret transition conditions received from the CVRE, determine the entry action, calculate the transitions and send the results back to the CVRE through the middleware.   The simulation langue receives the state to be transitioned to from the CVRE via a network to simulation adapter and sends messages back through the middleware to the CVRE via a simulation to network adapter.

The middleware is responsible for transferring the condition of the CVRE, which is the state requested to enter, to the simulation language and transfer the calculated action from the simulation language to the CVRE. In order to communicate with the CVRE and the simulation language, the CVRE to network, network to simulation, simulation to network and network to CVRE communication adapters behave as previously described. Figure 46 shows the state diagram of the specialized client-server architecture and Table 5 shows a description of the numbered specialized components within the CVRE, simulation language, and middleware.

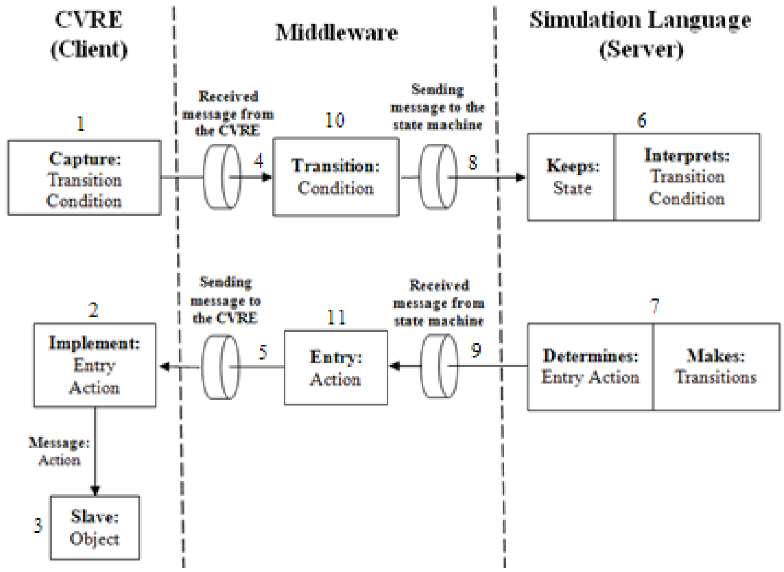


Figure 46. Hypothesized diagram of the communication protocol.

Table 5

Hypothesized List of Specialized Components within the CVRE, Simulation Language and Middleware

CVRE	Simulation Language	Middleware
• (1) CVRE Event Triggers • (2) Primary Event Adapter • (3) Secondary Event Adapter • Adapters: - (4) CVRE → Network - (5) Network → CVRE	• Server-Side State Machine - (6) State keeper and next state determination - (7) Action and transition update • Adapters - (8) Simulation → Network - (9) Network → Simulation	• (10) Condition Transfer • (11) Action Transfer

The developed communication protocol provides the guidelines for implementing any given simulation in a CVRE. In other words, the communication protocol is like a paint-by-number picture where the finite components, when executed correctly, make a masterpiece. Figure 46 provides the overall picture of how the simulation is to be executed in the CVRE. To fill in the communication protocol diagram with the information pertinent to the simulation to be executed in the CVRE requires developing the transition table and diagram. The purpose of the transition table and diagram is to help determine the components, as numbered in Table 5, that are needed for implementing the given simulation in the CVRE, simulation language, and middleware. The basic structure of a transition table is to list the possible states that can exist and what states they can move into which are the basic elements of a state machine. On the other hand, a state diagram provides a visualization of a state machine. Figure 47 provides an example of the state transition table (left) and state diagram (right) of a simulation with n states and n transitions.

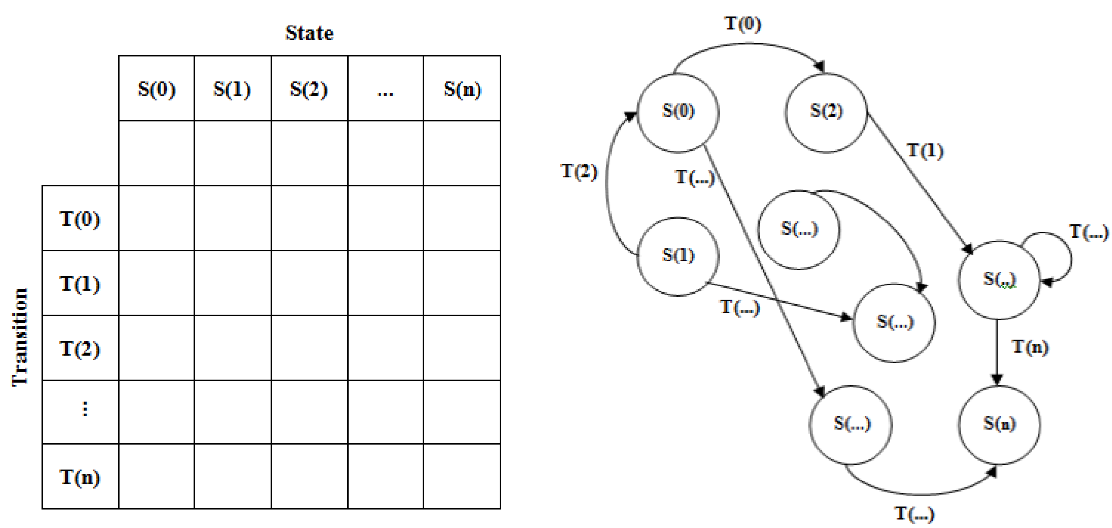


Figure 47. An example of a transition table (left) and state diagram (right) with n states and n transitions.

To get a better idea how a communication protocol is developed an easy simulation to consider is when that allows a user to open or close a door. Before considering what is needed in the CVRE, the basic functions of the door are open and close, which can be represented as a 0 and 1 respectively. From this information, a state transition table and state diagram for this simulation can be generated as seen in Figure 48.

Transition \ State	Open (0)	Close (1)
Open (0)	"Nothing"	"open door"
Close (1)	"close door"	"Nothing"

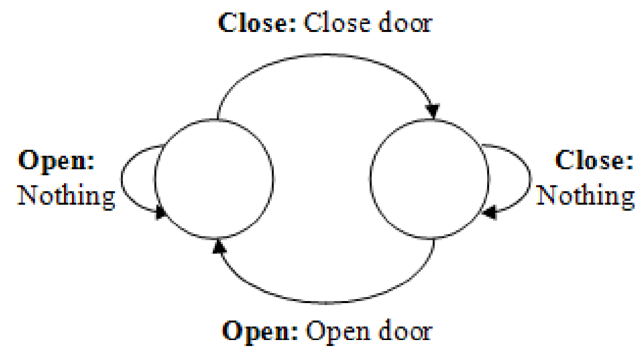


Figure 48. Transition table and state diagram for simulating a light that can be turned on and off.

From this information the state machine can be completed by filling in the requirements of the CVRE, middleware, and the simulation language. To complete this simulation, a button in the CVRE is needed that can capture the user pressing a button to trigger one of the two states, open or close, a means to tell the door what to do when the next state is calculated, and an object that resembles a door. The middleware needs means to receive the transition from the CVRE and forward it to the simulation language. In addition, after the next state is calculated, the means are also needed to receive the instructions on what needs to be done from the simulation language and forwarding this entry to the CVRE. Lastly, the simulation language needs to be able to keep the current state and be able to determine how to make the requested transition of the CVRE. After determining what the next state is, the simulation language also needs the ability to send the directions on how to transition to the next state back to the CVRE through the middleware. The completed state machine diagram is shown in Figure 49.

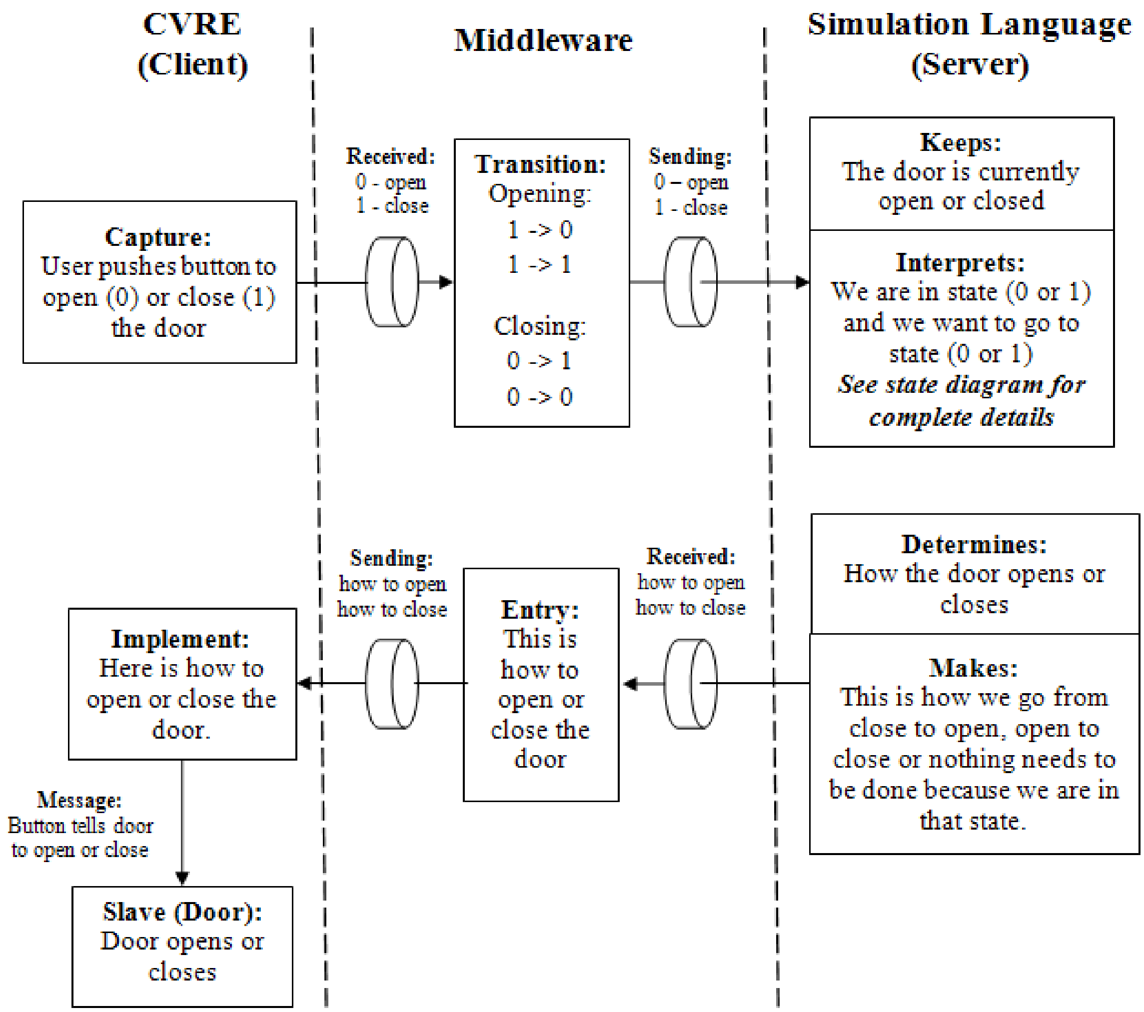


Figure 49. Completed state machine diagram for the simulation of lights on and off.

Once the communication protocol diagram is complete, the next step is to begin implementing each specialized component. Based on experience, the best approach for implementation is to begin by building a small test between all three components. The goal for implementing a small test case is to make sure that the languages chosen for the CVRE, middleware, and simulation language can be connected in order to communicate and determine the format of messages that need to be communicated between the three components. Once this small test case is implemented, the next steps include implementing the components in the simulation language, CVRE, and middleware in that order of importance. The reason for implementing the simulation first is because this is where the state machine resides. Once the state machine is implemented, the triggers that need to reside in the CVRE can be determined and thus the next step becomes implementing the components within the CVRE. At this point in the implementation of the CVRE, the goal is to develop the visualization of the CVRE and objects that can be manipulated by the user in order to trigger an event. After the elements and components are developed in the CVRE, the next step is to implement the Middleware. The middle ware needs to connect to the CVRE and the Simulation language as well as be able to receive a message from either the CVRE or Simulation and forward it to the correct recipient. The last step is to revisit the CVRE and implement the reaction, known as the simulation, for the incoming messages from the simulation language.

CHAPTER VI

TEST CASE

As indicated in Chapter I, the objective of this dissertation was to develop a robust and scalable communication method that connects external simulation languages with client-server Collaborative Virtual Reality Environments. The selection of a case study (Chapter IV) allowed a communication protocol to be hypothesized (Chapter V). In order to determine if this hypothesized communication protocol worked correctly, a test case was needed. This chapter provides an overview of the selected test case, preliminary design, desing using the proposed communication protocol from Chapter V, code implementation and testing of a simulation in the CVRE using the hypothesized methodology for developing a communication protocol between CVREs and simulation languages.

Selected Test Case

The test case selected was a project for Professor Hal Johnston, a Construction Management Professor at California Polytechnic State University. Professor Johnston is the owner and developer of “The Building Industry Game” (BIG). BIG is a teaching simulation which places students as managers of a construction company. Students who play BIG make decisions which emulate the thought process of real construction managers as they consider the price, method, labor, timing, and so forth. BIG fills a critical hole in these students’ education by providing hands on experience with the decisions they will make every day as a construction manager. The game itself is an entirely web-based text interface as seen in Figure 50 which does not provide a 3D visualization for the decisions made by students.

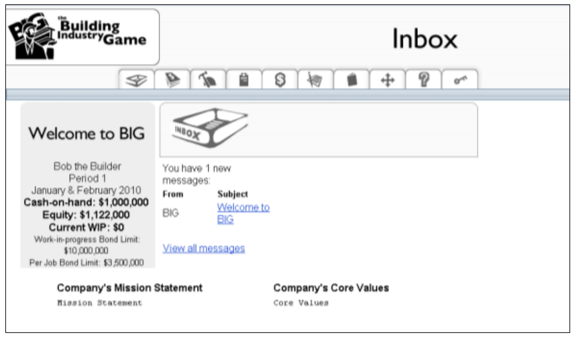
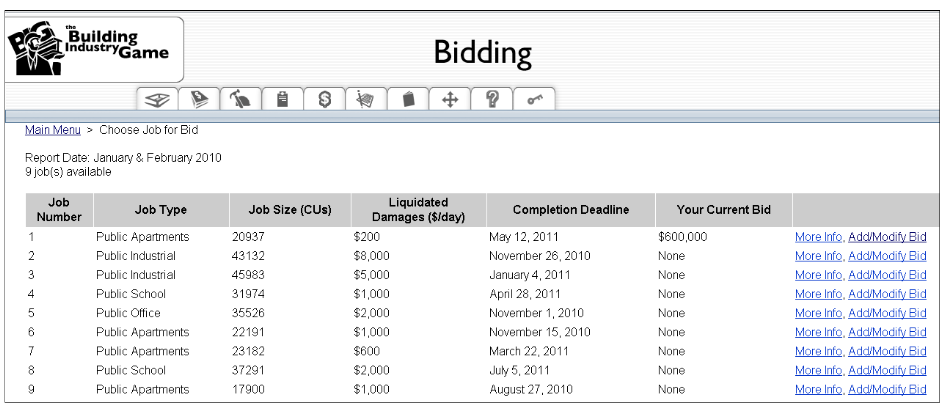


Figure 50. Home screen as seen in BIG.

Due to the complexity of BIG, the section determined to benefit from a 3D visualization was illustrating the relationship between the job size, type and method of excavation selected. The focus of the test case for the purpose of testing the communication protocol was to determine a better method to show how the size and type of a selected job affected the excavation method selected. In the original version of BIG, students are given the bidding information on all available jobs for which they can place a bid. The information as displayed to the students can be seen in Figure 51.



Job Number	Job Type	Job Size (CUs)	Liquidated Damages ($/day)	Completion Deadline	Your Current Bid	
1	Public Apartments	20937	$200	May 12, 2011	$600,000	More Info, Add/Modify Bid
2	Public Industrial	43132	$8,000	November 26, 2010	None	More Info, Add/Modify Bid
3	Public Industrial	45983	$5,000	January 4, 2011	None	More Info, Add/Modify Bid
4	Public School	31974	$1,000	April 28, 2011	None	More Info, Add/Modify Bid
5	Public Office	35526	$2,000	November 1, 2010	None	More Info, Add/Modify Bid
6	Public Apartments	22191	$1,000	November 15, 2010	None	More Info, Add/Modify Bid
7	Public Apartments	23182	$600	March 22, 2011	None	More Info, Add/Modify Bid
8	Public School	37291	$2,000	July 5, 2011	None	More Info, Add/Modify Bid
9	Public Apartments	17900	$1,000	August 27, 2010	None	More Info, Add/Modify Bid

Figure 51. Bidding screen as seen in BIG.

Selecting the “Add/Modify Bid” button on this screen navigated students to a different screen, as seen in Figure 52, where students selected what type of method they wanted to use on a particular job.

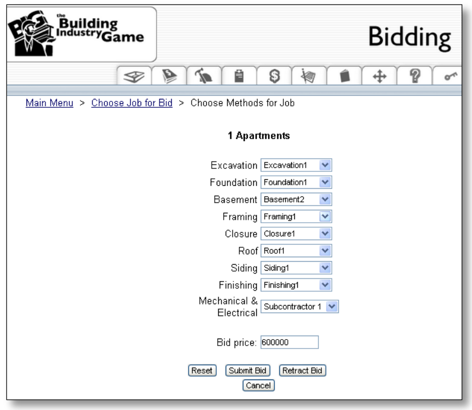


Figure 52. Screen navigated to by selecting the "Add/Modify Button."

After their selection was made, students continued bidding on other jobs. When the students selected the method of excavation it would have been beneficial to have a 3D visualization for the job size, type and excavation method selected so that the students could see the amount and type of equipment or personnel required to complete the selected job. Selecting this project as a test case would not only provide a 3D visualization in a CVRE for students playing BIG but also serve for testing the hypothesized communication protocol.

Preliminary Design

Once it was determined what visualization would be implemented in BIG, the next step completed was the project design. There were two objectives for designing a visualization in BIG which included how students' answers combined with the project information would impact the visualization and how the designed project would be mapped to the methodology for linking this simulation with the CVRE.

To develop a visualization that would reflect the student's selections in combination with the project information required obtaining a better understanding of how the job type, size and excavation method would affect a real-world excavation project. In order to obtain an insight to this matter, several conversations were held with Professor Johnston until a realistic representation was derived. Based on our discussions, it was interpreted that a job type related to an excavation for a particular building, which in BIG included an apartment, school, office, hospital and an industrial building. Each of these jobs ranged in size which implied the bigger the building, job size, the more expenses required to hire the excavation crew. Thus, it seemed best to represent the job types as different excavation plans and use varying amounts of people and/or equipment to represent the job size and excavation method.

In order to represent the excavation plans, four sets of plans were obtained including one for the apartment, school, office, hospital and industrial building as illustrated in Figure 53.

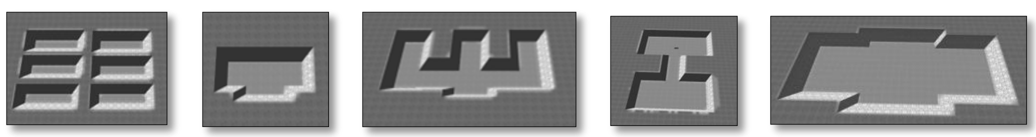


Figure 53. Types of plans to be excavated. From left to right: apartment, school, office, hospital and industrial.

In order to determine the amount and type of equipment or personnel required to complete the project required determining how to combine the job size and the method selected by the students. It was determined that the best way to represent the job size was by dividing it into ranges such that all job sizes would fit into one range bracket. The ranges were set as the following:

- Range 1: a job size less than 20,000.
- Range 2: a job size greater than or equal to 20,000 and less than 40,0000.
- Range 3: a job size greater than or equal to 40,000 and less than 80,000.
- Range 4: a job size greater than or equal to 80,000.

After the ranges were set, the next step was to determine how the excavation method would come in to affect. The choices from which a student could select an excavation method included excavation1, excavation2, excavation3, excavation4, or subcontractor. The chart as seen in Figure 54, which is also available to students in BIG, was utilized to help determine how the excavation method selected by the students would impact the results.

	Method #1				Method #2				Method #3				Method #4				Method #5			
	Labor ($)	Material ($)	Subcontractor ($)	Days	Labor ($)	Material ($)	Subcontractor ($)	Days	Labor ($)	Material ($)	Subcontractor ($)	Days	Labor ($)	Material ($)	Subcontractor ($)	Days	Labor ($)	Material ($)	Subcontractor ($)	Days
Excavation	47443	43130	0	67	77634	51756	0	44	77634	94886	0	35	129390	125077	0	24	0	0	357979	17
Foundation	77640	62112	0	92	139752	77640	0	58	129400	129400	0	49	243272	155280	0	32	0	0	522776	24
Basement	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Framing	103520	82816	0	129	191512	108696	0	80	232920	181160	0	58	346792	217392	0	43	0	0	791928	30
Closure	181160	51760	0	129	191512	181160	0	80	248448	217392	0	64	465840	326088	0	38	0	0	947208	32
Roof	113872	113872	0	161	186336	155280	0	107	232920	284680	0	71	414080	388200	0	46	0	0	983440	38
Siding	155280	129400	0	161	326088	170808	0	92	465840	103520	0	80	517600	403728	0	49	0	0	1066256	43
Finishing	181160	207040	0	215	258800	414080	0	129	465840	207040	0	92	414080	647000	0	58	0	0	1361288	46
Mech/Elec	0	0	504595	495	0	0	698670	323	0	0	1164450	194	0	0	1707860	138	0	0	2328900	107

Figure 54. Table to indicate labor cost for each method of excavation.

The pertinent information in this chart is the labor cost for each of the various excavation methods. Using this intuition in combination with the interpreted conversations with Professor Johnston, it was determined that the basic method, “Method #1,” would include people and increasing methods (Method #2 ... 5) would gradually add more people and at certain levels begin adding heavy machinery such as excavators, backhoes, etc. Once the means for determining the impact of the method selected by the students and the job size was determined, it was necessary to determine how they would combine to produce one result. Since the job size indicated the complexity of the job and the excavation method selected indicated the amount and type of machinery or people, the best way to combine the two was to increase the number and type of machinery based on the particular job size. In other words, the bigger the job size, the more machinery and/or people based on the excavation method selected. Once it was determined how the student’s answers combined with the project information would impact the visualization, the next objective was to design the ideas using the hypothesized methodology of this dissertation, as described in Chapter V.

Design Using Proposed Communication Protocol

In order to determine if the generalized communication protocol hypothesized in Chapter V worked as proposed, the test case was designed using the steps outlined for the proposed communication protocol. The steps first required designing the state diagram and transition table then utilizing both components to derive a state machine diagram, the communication protocol, which would provide a recipe for connecting the simulation language with the Collaborative Virtual Reality Environment (CVRE).

As outlined in the test case, Chapter IV, this project allowed students to select a type of project for which they wanted to make a bid and a method for excavating the project. In addition to the information provided by the students, the other piece of information evaluated included the job size of the project selected by the student. Using these three pieces of information, a visualization was to be executed in a CVRE where the type of project was illustrated as a particular excavation plan and the job size and method were illustrated through varying amounts and type of equipment. In order to provide an illustration for this process, the state diagram, Figure 55, was developed.

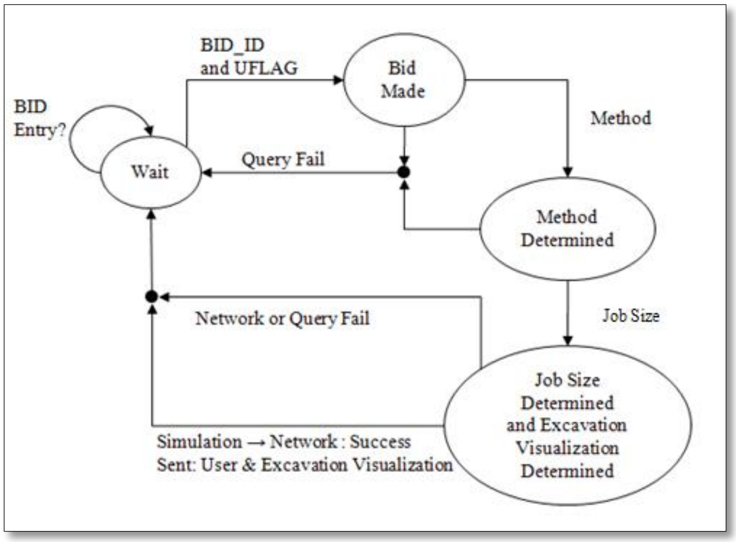


Figure 55. Test case state diagram.

This diagram provides a visual to provide insight for how the code should be implemented to execute the desired visualization. Once the state diagram was complete, the next step was to determine what would trigger a transition between two states and what the action would be if that event was triggered.

The transition table provided a more detailed listing of the components needed to implement the simulation in a CVRE. In order to create the transition table, all possible states were listed down the first column of a table and all possible transitions listed across the first row of the same table leaving the top right cell empty. In the remaining blanks the event to take place if a state was visited by that particular transition. Table 6 shows the completed transition table for the state diagram in Figure 55.

Table 6

Test Case Transition Table

	Bid Database Entry	User, BID_ID, UFLAG	Method	CEU	Simulation -> Network Success Sent: User and Excavation Visualization
Wait	Query for user, Bid_ID, and user flag. Transition to Bid Made	User has made a bid. Query for the user, Bid_ID, and UFLAG.	-	-	-
Bid Made	-		Query for CEU	-	-
Method Determined	-	-	-	Determine the excavation to be made	-
CEU and Excavation Visualization Determined	-	-	-	-	Feed message to simulation adapter and transition to wait stage

While the state diagram and transition table both provided insight on how to implement the simulation, more information was needed to develop the communication between the CVRE and the simulation language which could be obtained through developing the state machine diagram.

Before developing the state machine diagram, the three main components of the diagram, the CVRE, simulation language and the middleware were determined to be O3D, BIG, and PHP, respectively. Before information could be inserted into the state machine diagram research was done on each of the individual components to determine how they worked. In order to complete this research and develop the project, permission to access BIG had to be granted by Professor Johnston since BIG was hosted on a server at Cal Poly. Because Professor Johnston did not have any manuals on the code structure or the design of BIG, a lot of work was done to determine how BIG obtained, stored and processed information during game play. Since BIG was written using Java Server Pages (JSP) any information provided or stored on the page could be obtained by using a Java Script function, document.getElementsByName(). To pass this information to O3D, research showed using PHP to formulate a JavaScript file was the best method because O3D utilizes JavaScript files to drive the simulations. Once the study was completed, the state machine diagram as seen in Figure 56 was completed and the test case could be implemented.

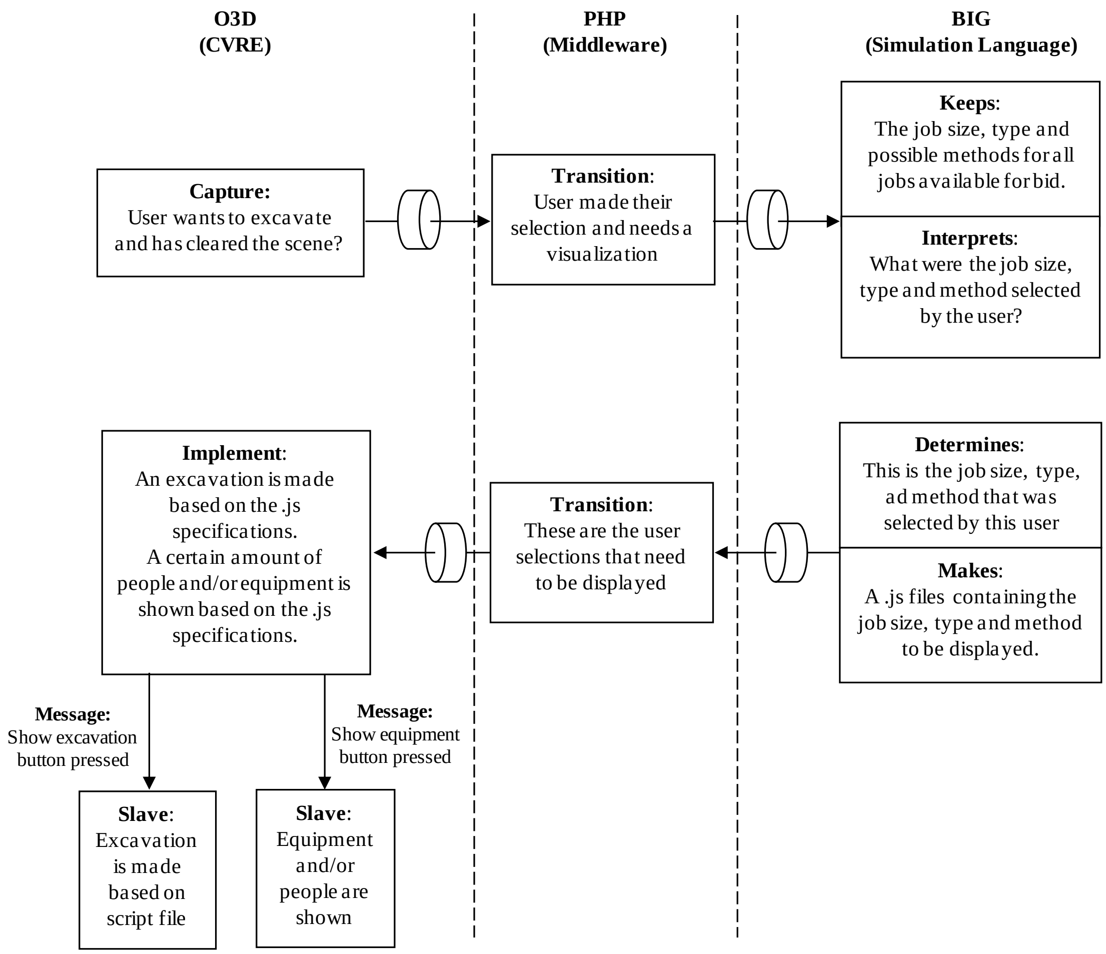


Figure 56. Test case state machine diagram.

Code Implementation

In order to implement the test case, the state machine diagram was translated into code. Implementing the project through code required, adding code to BIG, writing the PHP code to communicate between BIG and O3D, and writing the code in O3D to show the simulation. Because BIG was already implemented in code to execute the game, only two segments of code needed to be added. Those segments included one for grabbing the job size, type and method selected by the student and one for calling a PHP function to handle sending the obtained information to O3D. The code added to BIG as seen in Figure 57 utilizes the document.getElementsByName () function to grab the information from the Java Server Page and then formats the data into a string passed to a PHP file, write.php, on the O3D server.

```
<SCRIPT TYPE = "text/javascript">
    function processInformation()
    {
        var v = document.getElementsByName("methodForActivity[1]")[0].value;
        var i = document.getElementsByName("methodForActivity[1]")[0].options[v-1].text;
        var s = <%= request.getParameter("jobSize")%>;
        var p = <%= request.getParameter("jobID")%>;
        var t = <%= request.getParameter("jobType")%>;
        var popup = window.open
        ('http://bigcat.cs.usm.edu/~lduck/cycle7/scripts/write.php?s='+"var jobSize = "+s+
        "; var jobID =  " + p+ "; var jobType = " + t + ";" + "methodSelected = '" + i +
        "' ;", "File wirtten","menubar=1, resizeable=1, width=350, height=250");
        popup.close();
    }
</SCRIPT>
```

Figure 57. Code added to BIG.

After the code was added to BIG, the next step involved writing the PHP code to be located on the O3D server that would write the information passed from BIG to a JavaScript file. The PHP file grabbed the string passed from BIG and wrote it to a page called data.js which could be read and modified as needed in O3D. Figure 58 shows the write.php file located on the O3D client and Figure 59 shows the data.js file which is created on O3D from the call to the write.php by BIG.

```
    $mystring = $_GET['s'];
    $filename = "data.js";
    if(!is_writeable($filename)){
        echo 'The file is not writeable!!';
    }
    $file = fopen($filename, 'w');
    fwrite($file, $mystring);
    fclose($file);
?>
```

Figure 58. PHP file to pass information to O3D.

```
var jobSize = 20937; var jobID =  1;
var jobType = 1; var methodSelected = 'Excavation1' ;
```

Figure 59. The .js file created from the PHP file in Figure 7.

Once this information was obtained on the O3D client, the next step was to implement the simulation in O3D.

Implementing the code in O3D required studying an existing simulation in that CVRE in order to determine how to utilize the existing commands to show the excavation as well as the number and type of equipment for the project being considered. From this study, it was determined that network.send() commands would be used to send commands to the network in order to modify the environment based on the values in the JavaScript file. The network commands utilized were the terrain modifier, object creator, object locator, and object deletion. The purpose of the terrain modifier was to excavate the terrain based on a particular project design. The network.send() command for modifying the terrain, or making the excavation required the name of what was to receive the action, the action to be executed on the object, the x, y, and z location of the tile to be excavated, the x, y, and z dimensions of the are to be excavated, and the slope. On the other hand, to create an object such as an excavator, the network.send() command still needed an action, which was the model, a command index to signify an object needed to be created, an object ID to denote a certain object, and a url, which was the actual model. To place the object in the correct location, the network.send() command needed an action, which was a model, a command to signify the object described should be relocated, and a matrix to signify that the particular location of the object. Lastly, to delete an object, the network.send() command required an action, which was the model to be deleted, the command to signify the deletion of a model and the models ID. Due to the complexity of each network.send() command, the code implemented to handle the excavation and object manipulation can be seen in the Appendix. Using these commands in with combination with the ranges discussed in the design section of this chapter a visual as seen in Figure 60 was made so that no matter what information was provided by the user an excavation was made and some type and amount of equipment was shown.

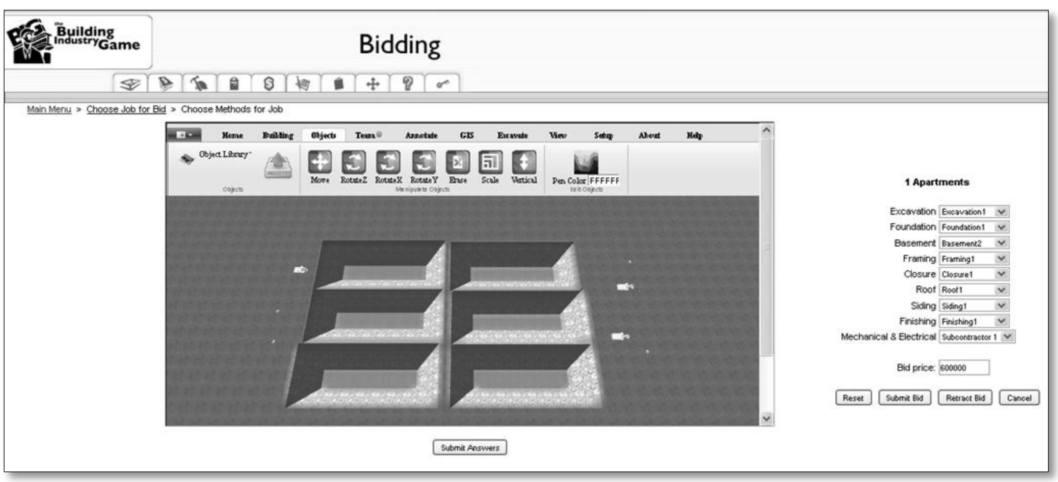


Figure 60. Visual of the simulation in BIG where the student selected an apartment project using method 1with a job size of 20,937.

Once the code was implemented, the next step was to let Professor Johnston test the simulation to ensure it would be beneficial to his students.

Testing

On December 16, 2010, a conference call was held with Professor Johnston in which he would be able to log into BIG and have a demonstration of what was created and then have time to experiment with the simulation on his own. After viewing the simulation Professors Johnston's first comment was "I had my doubts as to how this visualization would work but now I see how it can be extended throughout BIG". During the testing process no code failures occurred and the simulation went as expected producing the implemented simulations. As a result of feedback on the design of the simulation, Professor Johnston requested buttons be rearranged and some wording on the buttons changed. As an indicator of the successfulness of the implementation and testing of the simulation Professor Johnston prepared a letter of support which can be seen in Appendix B of this dissertation.

CHAPTER VII

RESULTS

The objective of this dissertation was to develop a robust and scalable communication method that connects external simulation languages with client-server Virtual Reality Environments. In order to develop this methodology a concrete case study was selected and implemented in order to formulate a hypothesized methodology and a test case was selected and implemented in order to finalize the hypothesized methodology. This chapter discusses the changes that were required to develop a finalized communication protocol and an example of how the methodology could be easily manipulated to add elements to a simulation in the CVRE.

Finalizing the Methodology

From the methodology derived in the concrete case study of this dissertation, the first step was to derive a state diagram and transition table to outline what needed to occur in the simulation. Once this was completed, the next step proposed was to determine the individual components necessary to implement the simulation by filling in the state machine diagram, Figure 61, using the list of specialized components in Table 7, to determine each individual task.

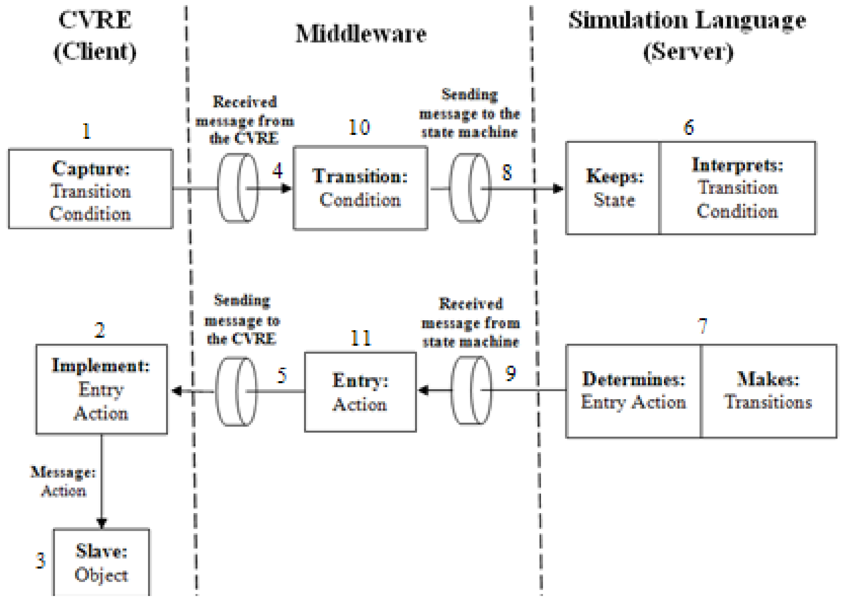


Figure 61. State machine diagram of the communication protocol where the numbers indicate the specialized component described in Table 7.

Table 7

Modified List of Specialized Components within the CVRE, Simulation Language and Middleware

CVRE	Simulation Language	Middleware
• (1) CVRE Event Triggers • (2) Primary Event Adapter • (3) Secondary Event Adapter • Adapters: - (4) CVRE → Network - (5) Network → CVRE	• Server-Side State Machine - (6) State keeper and next state determination - (7) Action and transition update • Adapters - (8) Simulation → Network - (9) Network → Simulation	• (10) Condition Transfer • (11) Action Transfer

Using this diagram, the simulation could be implemented and then testing began on the simulation in the CVRE. When using this methodology to develop the test case as outlined in Chapter VI, several critical elements were omitted from the hypothesized methodology. The addition of these elements would aid in minimizing the time required to implement the simulation and help in identifying the types of components and the order in which to implement them. The elements include understanding of the simulation to be implemented, determining what language(s) will be used to implement the Collaborative Virtual Reality Environment (CVRE), middleware and simulation language and how to pass messages between the client and the server. The remainder of this chapter provides the exact methodology for developing a communication protocol for visualizing simulation in a CVRE.

Before any design begins, the first and most important step in developing a simulation in a CVRE is to have a clear understanding of the simulation to be implemented. This requires meeting with the client to learn of the project, researching and meditating on their proposed and sometimes returning to the client for more clarifications. Once familiar with the simulation to be implemented, the next step is to design the transition table and state diagram. The purpose of this step is to turn the simulation to be implemented into a visual which will aid in determining how the simulation should be implemented. From this point, the next step is to determine the best CVRE for visualizing the simulation, the best simulation language to calculate the visual to be made in the CVRE, and the middleware for sending messages between the CVRE and simulation language. During this step, it is very important to make sure that both the CVRE and simulation can communicate with each other through a common middleware. In order to ensure that the communication is possible, a sub-step of this process is to implement a very small “hello_world” communication using the desired languages as seen in Figure 62.

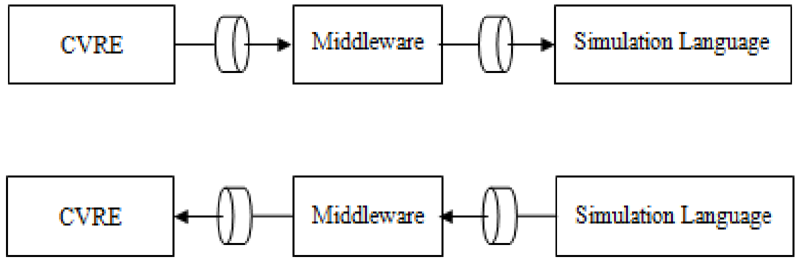


Figure 62. A diagram of the two-way communication to be established between the simulation language and CVRE.

The goal of this sub-step is to make sure that the languages selected are compatible, communication can exist between the CVRE and the simulation language, and an idea of how to design the simulation in the CVRE is grasped. In order to complete this process, code needs to be implemented to pass a message to the CVRE from the simulation and a different set of code to pass a message to the simulation language from the CVRE. Once the languages are determined with the assistance of the communication code, the next step is to design the state machine diagram. In order to design the state machine diagram a list of specialized components, Table 8, must be determined and placed in the state machine diagram as seen in Figure 63.

Table 8

Finalized List of Specialized Components within the CVRE, Simulation Language and Middleware

CVRE	Simulation Language	Middleware
• (1) CVRE Event Triggers • (2) Primary Event Adapter • (3) Secondary Event Adapter • Adapters: - (4) CVRE → Network - (5) Network → CVRE	• Server-Side State Machine - (6) State keeper and next state determination - (7) Action and transition update • Adapters - (8) Simulation → Network - (9) Network → Simulation	• (10) Condition Transfer • (11) Action Transfer

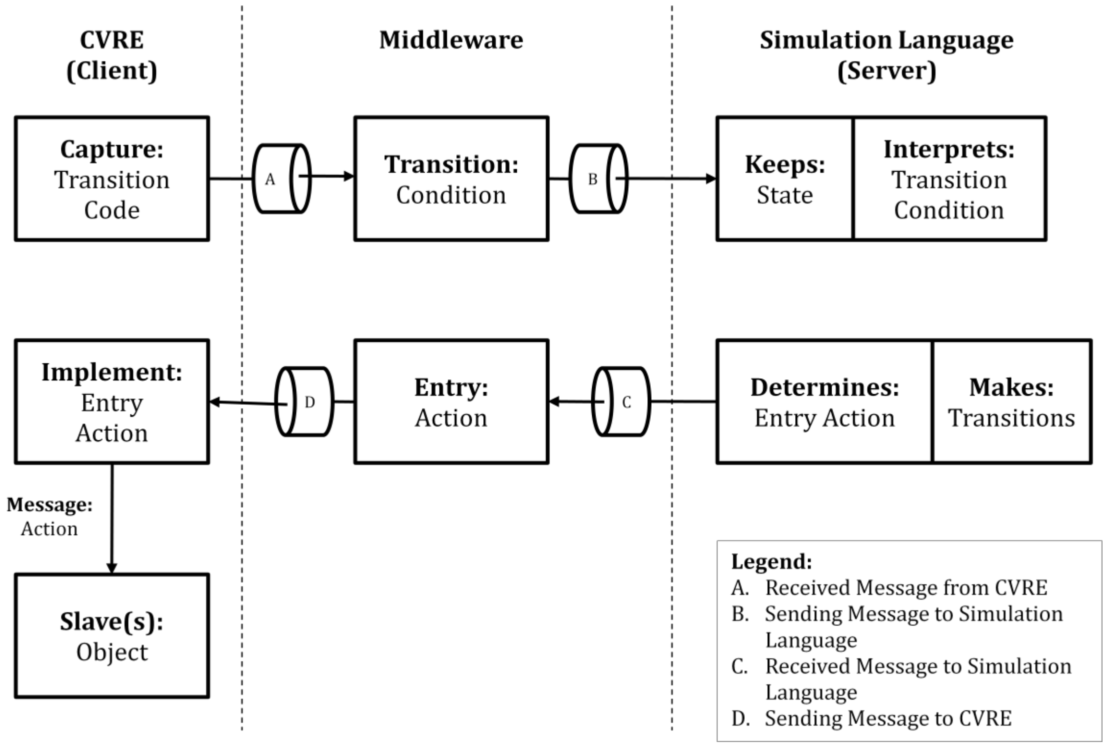


Figure 63. State machine diagram

This process is easy to do as the CVRE, middleware, and Simulation language have been selected from a previous step and an indication of how information is calculated and process. To fill in the chart, it is best to first determine what will trigger an action in the CVRE (Capture), what this action needs to tell the simulation language (Transition), what the simulation language will keep to make its judgment about moving to the next state (Keeps), and what the message received from the CVRE is asking and is this move possible(Interprets). The next elements that need to be determined are those that will determine what the simulation shown in the CVRE will be which requires determining what action should be made in the CVRE (makes), determine what message should be sent to the CVRE (determines), sending the action to be visualized to the CVRE (Entry), how to visualize the simulation in the CVRE (Implement), and what will be doing the actions in the CVRE(slave/slaves). After the state machine diagram is complete, a visual for how to implement the simulation in the CVRE is obtained and the simulation can be implemented.

In order to implement the simulation the state machine diagram can be used as a reference. In addition to this reference, the base communication between the CVRE and the Simulation language that was implemented earlier also acts as a starting point. From this, it is known how message will be received from and how messages can be sent to the CVRE and simulation language respectively. With this information in mind, the first step is to implement the code for the Simulation Language. Knowing what messages can come from the CVRE the cases can be built to handle these messages and determine what should occur and what should be sent to the CVRE. Once the simulation language is built, the next step is to implement the CVRE keeping in mind how message can be sent to the Simulation Language and what messages will be coming back from the simulation language. To implement the visual of the simulation on the CVRE side the first code to implement is that which captures the user prompting for a visual causing a message to be sent to the Simulation Language. When the prompt(s) has been implemented the next thing to code is how to visualize the possible messages coming from the Simulation Language. This step requires looking at how to represent the simulation visually in the CVRE using its properties to make the simulation come to life. Once the visual is complete, the last code to implement is the Middleware, which is made simple by referring to the test sample of two-way communication implemented at the beginning of the design process for the code and referencing the message to be sent to and received by the CVRE and Simulation Language. This step is very simple as it is known how to handle a message being received or how to send a message to either the CVRE or

Simulation Language using the two-way communication previously implemented. Completion of the Middleware results in a simulation visible in a CVRE that is ready to be tested and if necessary fine tuned until the desired simulation is perfected.

Manipulating the Methodology for Additional Simulation Features

The developed methodology can be quickly modified in order to add elements to a simulation. In order to provide an example of the simplicity of adding onto the existing simulation, adding a simulation onto the one as described in Chapter VI of this dissertation will be described. The simulation in Chapter VI utilized a piece of BIG in order to simulate an excavation based on the user input, job size, and type. For illustration purposes, the simulation to be added would be another element of big, which would be to simulate the foundation based on the user input, job size, and type.

As done in Chapter VI the first step is to design the state diagram. The only modifications to make to this particular state diagram are to replace the term Excavation with Foundation as highlighted in the Figure 64.

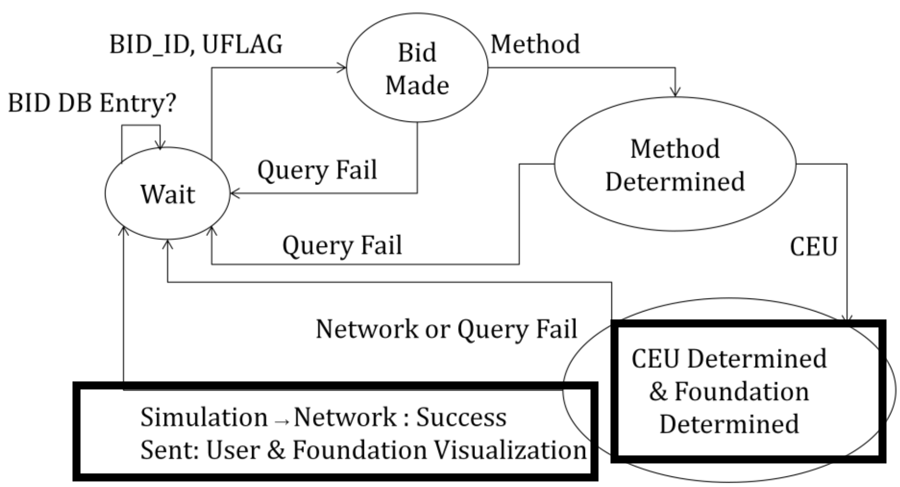


Figure 64. Changes to be made to the state diagram are highlighted in bold black.

After these changes are made to the state diagram, the next thing is to update the transition table. The changes that need to be made to the transition table are very minimal to reflect the changes in the state machine and can be seen in the table below. The last changes that need to be made to the communication protocol are simple and can be seen in Figure 65.

Table 9

Finalized Transition Table for Implementing the Foundation Simulation

	Bid Database Entry	User, BID_ID, UFLAG	Method	CEU	Simulation -> Network Success Sent: User and foundation Visualization
Wait	Query for user, Bid_ID, and user flag. Transition to Bid Made	User has made a bid. Query for the user, Bid_ID, and UFLAG.	-	-	-
Bid Made	-	-	Query for CEU	-	-
Method Determined	-	-	-	Determine the foundation to be made	-
CEU and foundation Visualization Determined	-	-	-	-	Feed message to simulation adapter and transition to wait stage

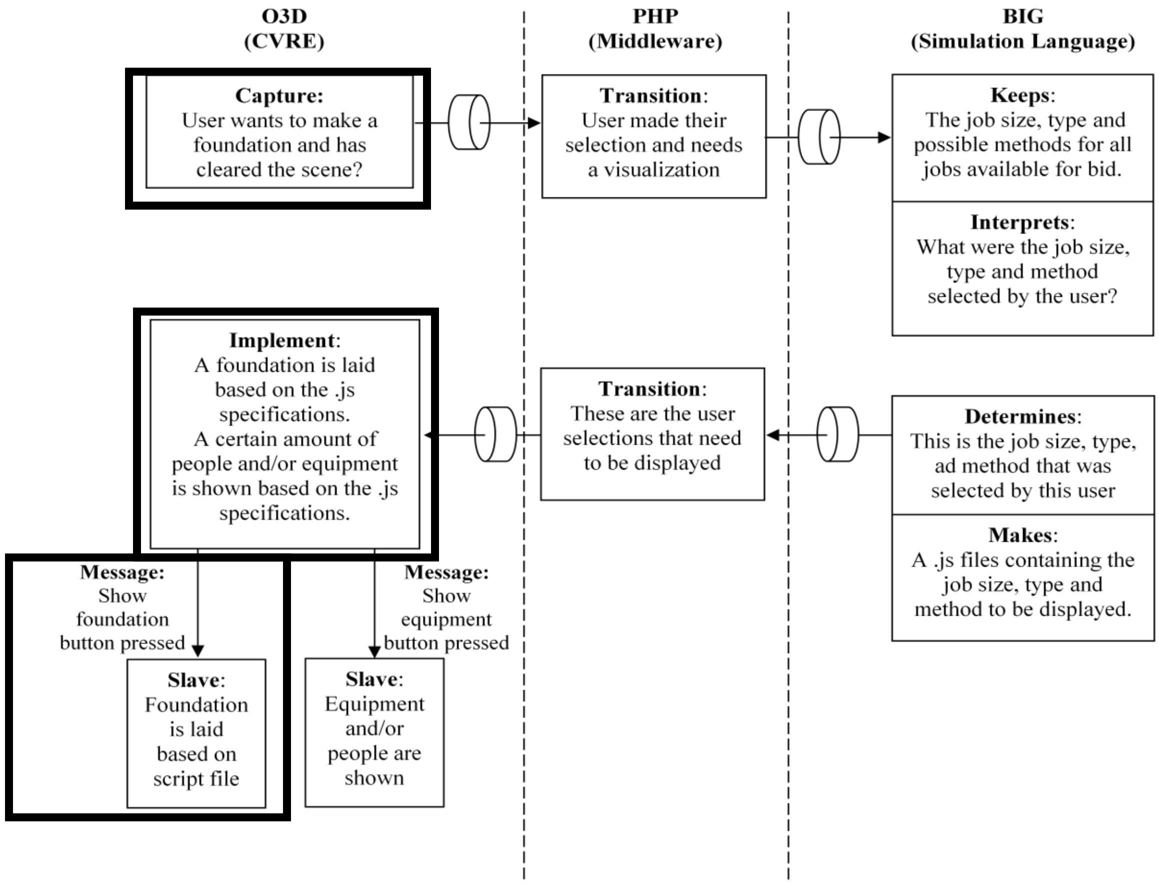


Figure 65. Changes to be made to the communication protocol are highlighted with bold black lines.

Although these changes are mute because the code has already been implemented to develop the first simulation, it is important to understand that the methodology can be easily manipulated to incorporate any simulation. The following figures show the major code segments used to implement each of the components of the communication protocol.

Starting with the Capture component, the user know wants to visualize a foundation. This requires noting the change in the capture component and then manipulating the existing code to add the new feature. In this case, a new button is required so a button is made with a different name that will call a different function and have a different image. Figure 66 highlights the differences in the capture component and the code.

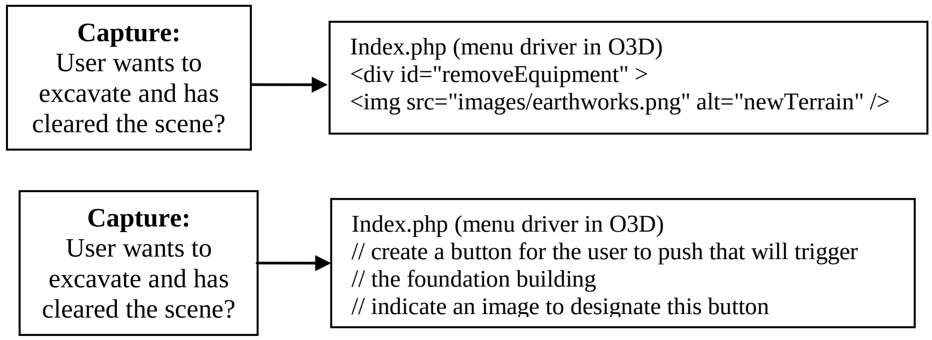


Figure 66. The capture component for the excavation simulation (top) and the foundation simulation (bottom).

The next change required is to the transition component and the code implemented to handle this action. In this instance, no changes are required for the transition component and only an addition to remove old foundations from the scene need to be implemented. However, as seen in the original code segment (top of Figure 67), existing code can be copied and minimally altered in order to provide this action.

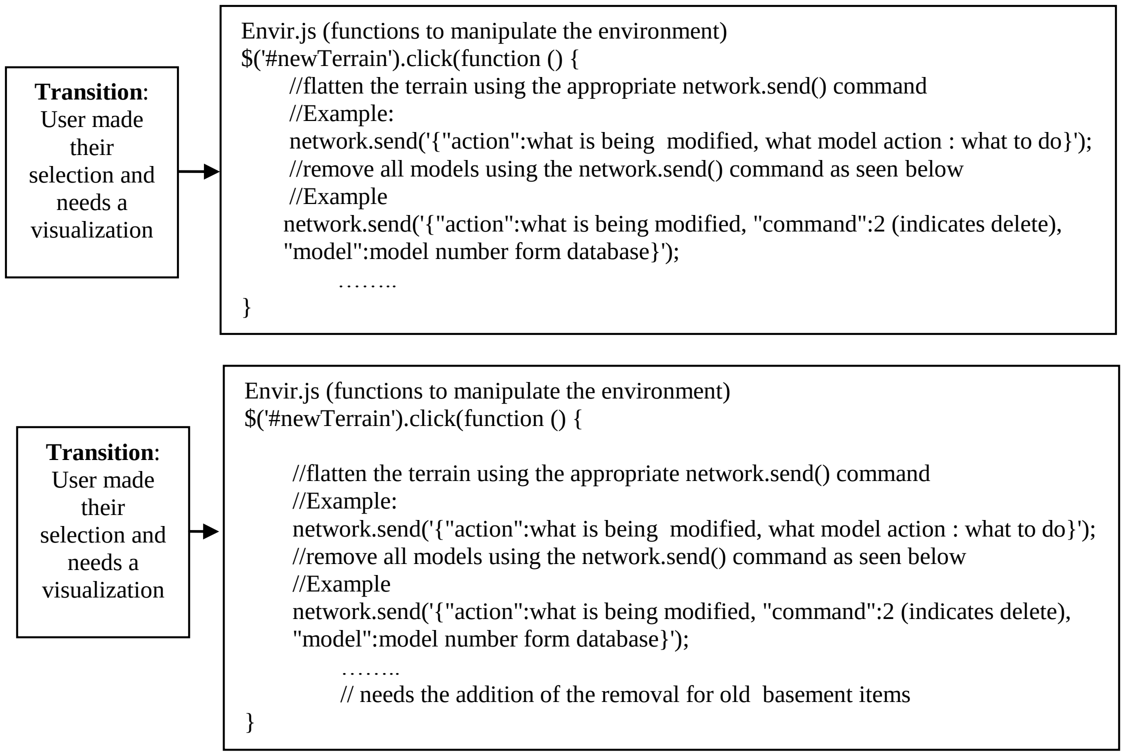


Figure 67. The transition component for the excavation simulation (top) and the foundation simulation (bottom).

After manipulating the transition component, the next component to modify is the keeps component. While the wording for the components does not need to be changed drastically, the developer still has to keep in mind that the method selected by the user is a foundation. Because this is a different method and the two methods selected need not be confused, a variable to hold the foundation method is needed in the code. This means the interpret component needs the code necessary to actually determine what the foundation method selected by the user would be. Figure 68 indicates the requirements.

Keeps: The job size, type and possible methods for all jobs available for bid.

Interprets: What were the job size, type and method selected by the user?

```
<SCRIPT TYPE = "text/javascript">
    function foo()
    var v = document.getElementsByName("methodForActivity[1]")[0].value;
    var i = document.getElementsByName("methodForActivity[1]")[0].options[v-1].text;
    var s = <%= request.getParameter("jobSize")%>;
    var p = <%= request.getParameter("jobID")%>;
    var t = <%= request.getParameter("jobType")%>;
    var popup =
    window.open('http://bigcat.cs.usm.edu/~lduck/cycle7/scripts/write.php?s='+"var
    job Size = "+s+ "; var jobID =  " + p+ "; var jobType = " + t + ";" +
    "methodSelected = '" + i + "' ;", "File wirtten", "menubar=1, resizeable=1, width=350,
    height=250");
    popup.close();
</SCRIPT>
```

Figure 68. The keeps and interprets component for the excavation simulation.

The next component that needs modifying is the determines component which does not have to be altered by the same rationale of the keeps component as seen in Figure 69. However, on the coding side the designer can either create a new file to write the results to or the value can be combined in with the existing string.

Determines: This is the job size, type, ad method that was selected by this user

Makes: A .js files containing the job size, type and method to be displayed.

```
<SCRIPT TYPE = "text/javascript">
    function foo()
        var v = document.getElementsByName("methodForActivity[1]")[0].value;
        var i = document.getElementsByName("methodForActivity[1]")[0].options[v-1].text;
        var s = <%= request.getParameter("jobSize")%>;
        var p = <%= request.getParameter("jobID")%>;
        var t = <%= request.getParameter("jobType")%>;
        var popup =
            window.open('http://bigcat.cs.usm.edu/~lduck/cycle7/scripts/write.php?s='+"var
            jobSize = "+s+ "; var jobID =  " + p+ "; var jobType = " + t + ";" +
            "methodSelected = '" + i + "' ;", "File wirtten", "menubar=1, resizeable=1,
            width=350, height=250");
            popup.close();
</SCRIPT>
```

Figure 69. The determines and makes component for the excavation simulation.

The next component to be modified is the transition component which in this case did not have any modifications even on the coding side as seen in Figure 70. The reason a change was not required related to the choice made in coding the makes component. If the designer desires a different file, the code segment in the figure below would have to be manipulated in order to reflect this change.

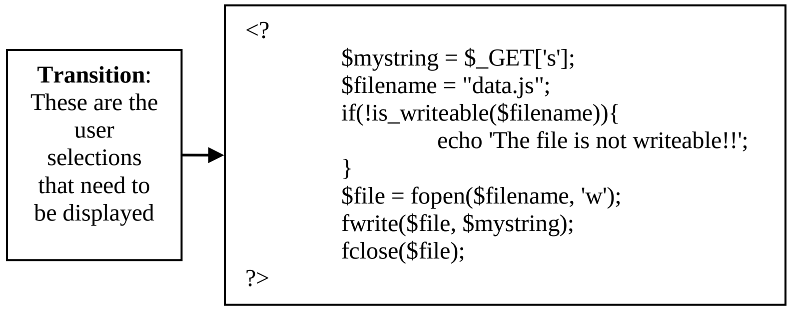


Figure 70. The transition component for the excavation simulation.

The next component to be modified is the implement component the only change in the component itself is the replacement of the term excavation with foundation as seen in Figure 71. Unlike the previous examples, the other element to change is not code as it is the file created from the transition component. This file might have a different name if a new file was created in the transition component code, otherwise it will just have an additional variable. The figures below shows the original implement component and the output file as well as the modified implement component and the new file.

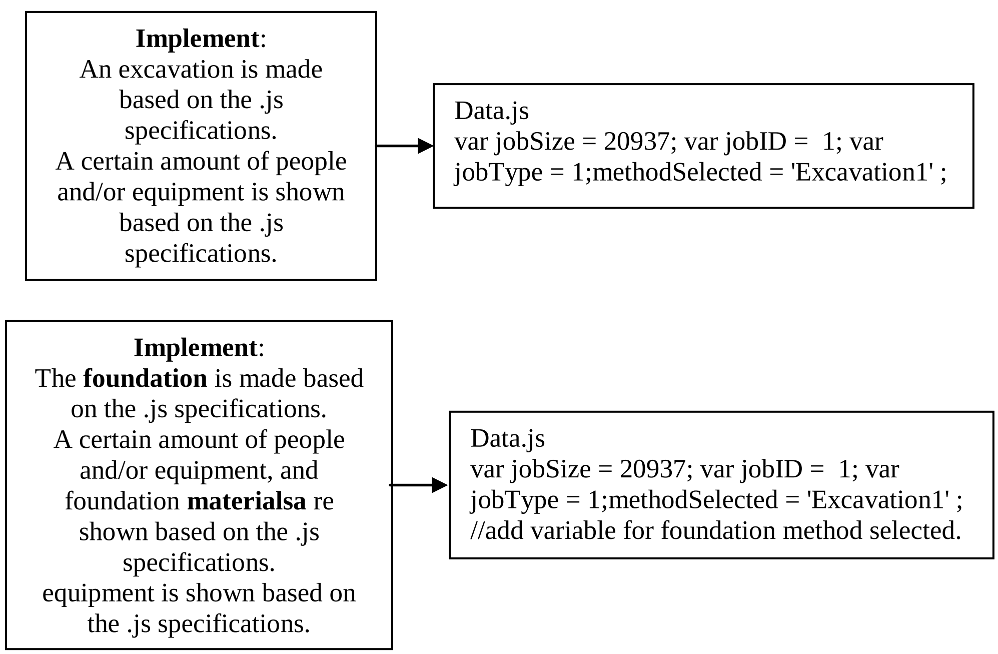


Figure 71. The implements component for the excavation simulation.

The last components to be implemented include the slaves and the code required to execute these components as seen in Figure 72. In the first slave component, the only change required is the replacement of the term excavation with foundation. The code will need to be updated to include what methods could be selected for the type of building foundation to be made which can quickly be done by modifying the existing code. Both the original and modified component and code can be seen in the figure below.

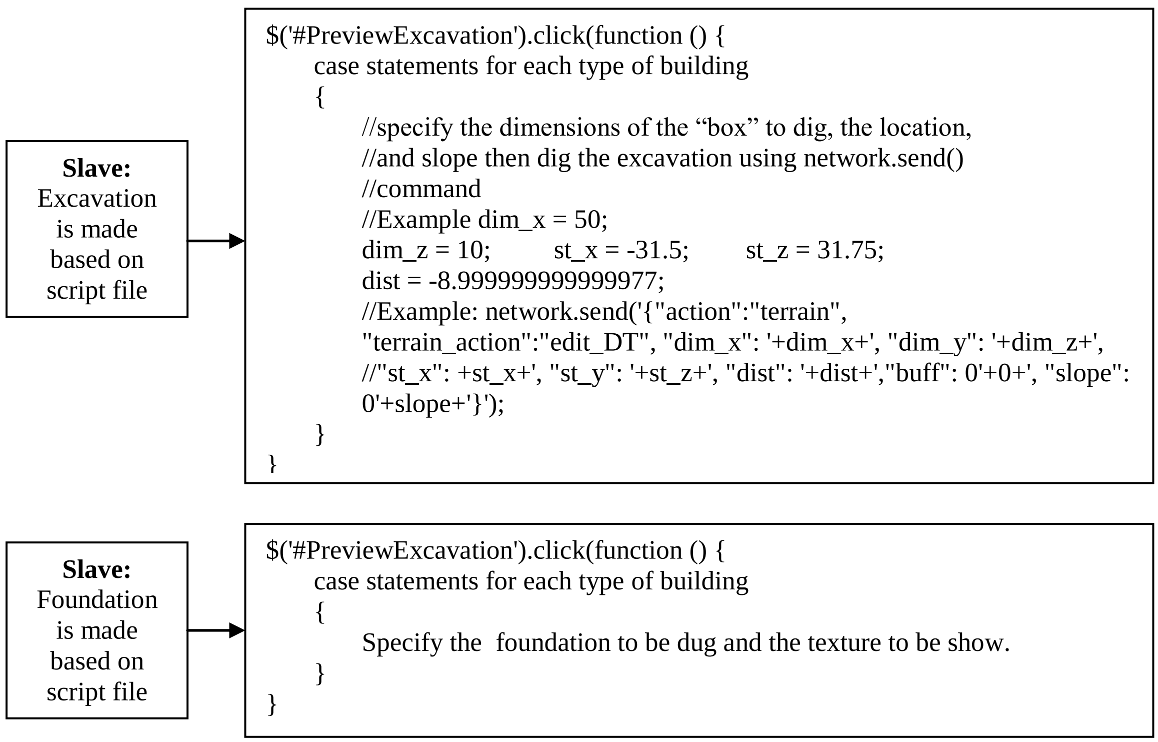


Figure 72. The first slave component for the excavation simulation (top) and the foundation simulation (bottom).

The last component to be modified in this case is the second slave component. Like previous examples the componenet requires minimal changes of the term excavation with foundation as seen in Figure 73. The code itself is modified to include cases for all combinations of job sizes and foundation methods selected as well as the code to make the actual foundation. The code can quickly be copied and tailored to fit the desired terms asthe model to be visualized will change from a certain type of equipment to perhaps an element of foundation.

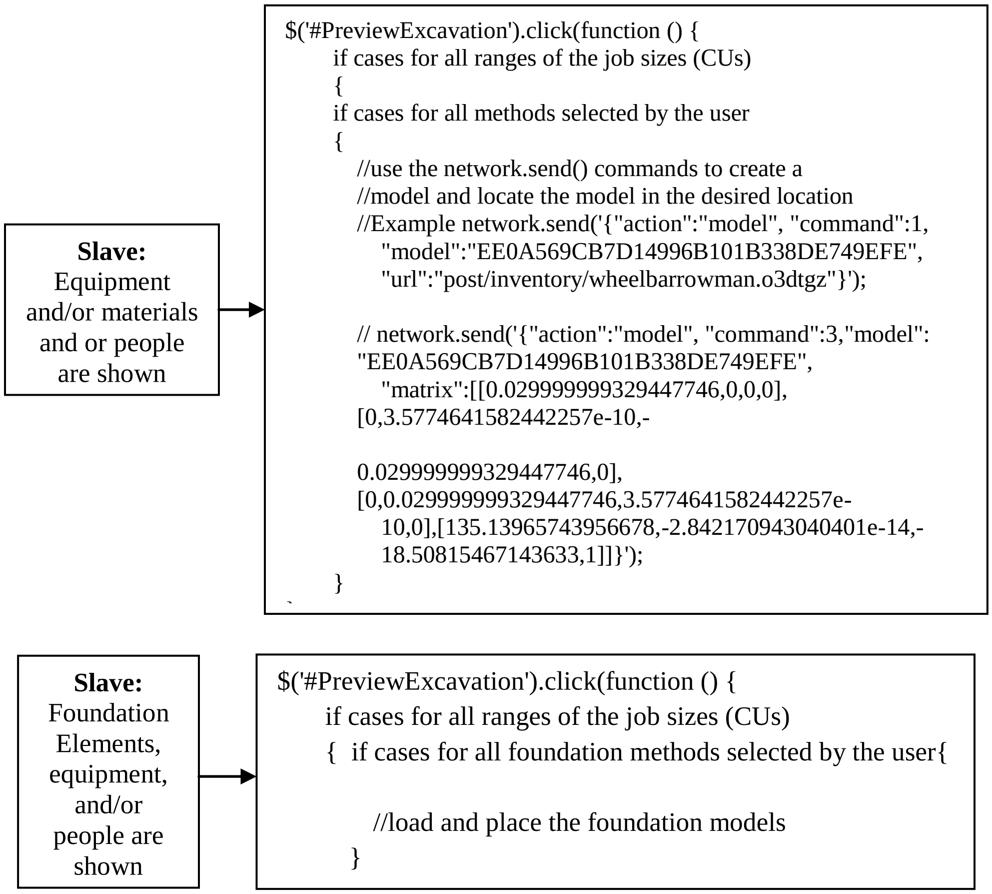


Figure 73. The second slave component for the excavation simulation (top) and the foundation simulation (bottom).

As seen in this chapter, the methodology developed from this dissertation will allow simulations in a collaborative virtual reality environment to be easily developed and extended to add onto the developed simulation with ease. However, as discussed in the next chapter, there are some limitations in this methodology and it could benefit from future research.

CHAPTER VIII

SUMMARY

The objective of this dissertation was to develop a robust and scalable communication method that connects external simulation languages with client-server Virtual Reality Environments. Chapter VI describes the methodology developed by using a concrete case study to design a hypothesized methodology and a test case to perfect the methodology. Before this methodology, implementing simulations in Collaborative Virtual Reality Environments (CVRE) were very difficult and lead to "spaghetti code." The result of this dissertation is the methodology of a communication protocol for developing simulations in CVRE. This methodology provides easy design and implementation of any simulation to be visualized in a CVRE and the ability to easily extend the simulation by filling in the state machine diagram which a result of the methodology. The remainder of this chapter discusses the limitations and future research and impact of this dissertation.

Limitations and Future Research

While this dissertation provides the methodology to develop a communication protocol for visualizing simulations in CVRE there are several limitations in this methodology. This section discusses the limitations including the dealing with multi-user, language compatibility, scalability, communication packet size, and methodology testing and discusses the future research in order to address these limitations as well as future research that could extend from this methodology

Multi-User

The first limitation deals with the aspect of multi-user interactions. While the methodology should support multi-users it has not been tested on a large scale. In order to address this issue, future research could be to incorporate the ability of multi-user interactions.

Language Compatibility

The second limitation has to do with language compatibility. Because the communication protocol itself requires the simulation language to communicate with the selected CVRE if the languages are not compatible and a means to communicate between the two does not exist, then the methodology will not work.

Scalability

The next limitation deals with the scalability of the simulation to be visualized in the CVRE and the number of users. While the complexity of the simulation to be visualized has been tested using large simulations, it has not been tested using very large scale simulations. In addition to the scalability of the simulation itself, in regards to the large scale simulation, the issue of multi-users while in the CVRE visualizing the simulation has not been tested. In order to address this, future research would design a simulation in CVRE using this methodology and test how many users can interact simultaneously in the CVRE while the simulation remained stable.

Communication Packet Size

In designing the communication protocol the developed methodology only considered passing small packets of data using the method of receiving the data and processing it one at a time. Thus, future research is needed in the communication packet size and method in order to ensure large packets of communication can be handled by this methodology. To address this limitation, larger packets of data would need to be passed through the communication protocol in order to determine its breaking point.

Methodology Testing

The last limitation is the assumption that this methodology could be used by others. While this may be true, others have not used the methodology to develop simulations in a CVRE. As part of future research it would be beneficial to distribute this methodology to others, allow them to design and implement the simulation in a CVRE and measure their success in terms of easiness, timeliness, and scalability.

Additional Future Research

While some future research can be done to enhance the designed methodology of this dissertation, other research can be done to extend the capabilities of the methodology. These include implementing simulations in CVRE for mobile devices, developing parallel communications for very large scale simulations, and using visual conflation to obtain data from different sources as a one-to-many form of communication.

Impact

As a result of this dissertation, a methodology is provided so that simulations in need of visualization can quickly and cost effectively be implemented in a CVRE for single users to visualize their interactions. From this methodology a better and more effective software development among simulation and training applications developed in CVREs. As a result of what the methodology can do and what it will result in, the impact of this dissertation will be more simulation and training software that would be better than those currently existing. While this impact effects only those in need of simulations the users of the simulations will results in a better trained work force equipped with better decision-making tools based on better and more widely available simulation and training software.

APPENDIX A

CONCRETE CASE STUDY

LavVIEW Code

Below is the code implemented in the simulation language LabVIEW to develop the test case simulation in RealXtend. This code was exported from LabVIEW using the Export to String method which generates a text file in the format of XML containing all the strings for implementing the code. In short, it is a capture of all the controls and indicators on the front panel which allows it to be imported into other LabVIEW applications.


```
<VI syntaxVersion=11 LVversion=8608001 revision=62     name="CompleteWorkingExcavation.vi">
<TITLE><NO_TITLE name="CompleteWorkingExcavation.vi"></TITLE>
<HELP_PATH></HELP_PATH><HELP_TAG></HELP_TAG>
<RTM_PATH type="default"></RTM_PATH>
<DESC></DESC><CONTENT>
<GROUPER><PARTS></PARTS></GROUPER>
<CONTROL ID=195 type="Tab Control" name="Tab Control">
<DESC></DESC><TIP></TIP><PARTS><DESC></DESC><TIP></TIP>
<PARTS><PART ID=92 order=0 type="Tab Caption">
<LABEL><STEXT><FONT predef=DLGFONT style='B' size=19 color=1000006>View1
</STEXT></LABEL></PART></PARTS><PART ID=8001 order=0 type="Array Index">
<CONTROL ID=87 type="Enum" name=""><DESC></DESC><TIP></TIP>  <PARTS>
<PART ID=12 order=0 type="Ring Text"><MLABEL>
<FONT predef=APPFONT color=100000C>
<STRINGS></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=DLGFONT color=100000C></STEXT></LABEL></PART>
</PARTS></CONTROL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=DLGFONT color=100000C>Tab Control</STEXT>
</LABEL></PART></PARTS><PAGE><DESC></DESC><TIP></TIP>
<CONTROL ID=95 type="Picture" name="picture 4">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this
output to any other <<B>>picture<</B>> input to add more drawing instructions to the    picture.</DESC>
<TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT style='B'>picture
4</STEXT></LABEL></PART>
</PARTS></CONTROL></PAGE><PAGE><DESC></DESC>
<TIP></TIP><CONTROL ID=95 type="Picture" name="picture 3">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this    output to
any other <<B>>picture<</B>> input to add more drawing instructions to the
picture.</DESC><TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT style='B'>picture 3</STEXT></LABEL></PART>
</PARTS></CONTROL></PAGE><PAGE><DESC></DESC>
<TIP></TIP><CONTROL ID=95 type="Picture" name="picture 5">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this    output to
any other <<B>>picture<</B>> input to add more drawing instructions to the
picture.</DESC><TIP></TIP><PARTS><PART ID=82 order=0 type="Caption"><LABEL>
```

<STEXT><FONT style='B'>picture 5</STEXT></LABEL></PART>
</PARTS></CONTROL></PAGE><PAGE><DESC></DESC>
<TIP></TIP><CONTROL ID=95 type="Picture" name="picture 2">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this output to any other <<B>>picture<</B>> input to add more drawing instructions to the picture.</DESC>
<TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT style='B'>picture 2</STEXT></LABEL></PART>
</PARTS></CONTROL></PAGE><PAGE><DESC></DESC><TIP></TIP>
<CONTROL ID=95 type="Picture" name="picture 6">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this output to any other <<B>>picture<</B>> input to add more drawing instructions to the picture.</DESC>
<TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT style='B'>picture
6</STEXT></LABEL></PART></PARTS></CONTROL></PAGE>
<PAGE_CAP_FONT><FONT predef=DLGFONT style='B' size=19
color=1000006></PAGE_CAP_FONT>
<PRIV><PAGE_CAPTIONS><STRING>View 1</STRING><STRING>View
2</STRING><STRING>View 3</STRING><STRING>View 4</STRING><STRING>View
5</STRING></PAGE_CAPTIONS></PRIV>
</CONTROL><CONTROL ID=95 type="Picture" name="picture 7">
<DESC><<B>>new picture<</B>> is the picture that contains the new image. You can wire this output to any other <<B>>picture<</B>> input to add more drawing instructions to the picture.</DESC>
<TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT style='B'>picture
7</STEXT></LABEL></PART></PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Correct Soil Type? ">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL><FONT
size=27><STRINGS><STRING>OFF</STRING><STRING>ON</STRING><STRING>OFF</
STRING><STRING>ON</STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT style='B' size=21>Correct Soil
Type? </STEXT></LABEL></PART>
</PARTS></CONTROL>
<LABEL><STEXT><FONT style='B' size=24>What is the soil type described in the bottom <LF>left
figure?</STEXT></LABEL>
<CONTROL ID=79 type="Boolean" name="TypeA">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>Type
A</STRING><STRING></STRING><STRING>Type
A</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>TypeA</STEXT></LABEL></PART>
</PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="TypeB">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Type B</STRING><STRING></STRING>
<STRING>Type B</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>TypeB</STEXT></LABEL></PART>
</PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="TypeC">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Type C</STRING><STRING></STRING>
<STRING>Type C</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>TypeC</STEXT></LABEL></PART>

</PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Strata">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Layered Strata</STRING><STRING></STRING>
<STRING>Layered Strata</STRING><STRING></STRING></STRINGS>
</MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Strata</STEXT></LABEL></PART> </PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Rock"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL><FONT style='B' size=20><STRINGS>
<STRING>Stable Rock</STRING><STRING></STRING>
<STRING>Stable Rock</STRING><STRING></STRING>
</STRINGS></MLABEL></PART> <PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Rock</STEXT></LABEL></PART>
</PARTS></CONTROL> <CONTROL ID=79 type="Boolean" name="stop">
<DESC>Stop communication</DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT predef=SYSFONT color=FF0000><STRINGS>
<STRING>STOP</STRING><STRING>STOP</STRING>
<STRING>STOP</STRING><STRING>STOP</STRING>
</STRINGS></MLABEL></PART> <PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>stop</STEXT></LABEL></PART> </PARTS></CONTROL>
<CONTROL ID=81 type="String" name="Input Line"><DESC>Text to send</DESC>
<TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>Input Line</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String" name="You Typed:">
<DESC>History of your dialog</DESC>
<TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL><STEXT>
</STEXT></LABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>You Typed:</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String" name="Their Response:">
<DESC>History of thier dialog</DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>Their Response:</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String" name="substring before match">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT size=16>substring before match</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<CONTROL ID=80 type="Numeric" name="Port"><DESC>TCP port</DESC><TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>Port</STEXT>
</LABEL></PART></PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Rock 2"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING>Get Bid Plan and Soil Type Information</STRING>
<STRING></STRING><STRING>Get Bid Plan and Soil Type Information</STRING>
<STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Rock 2</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=91 type="Path" name="UDL file path 5">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>

<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>UDL file path 5</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="absolute"></PATH></DEFAULT></CONTROL>
<CONTROL ID=79 type="Boolean" name="Simple Slope">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>Simple Slope</STRING><STRING></STRING>
<STRING>Simple Slope</STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Simple Slope</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Simple Bench">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Simple Bench</STRING><STRING></STRING>
<STRING>Simple Bench</STRING><STRING></STRING>
</STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Simple Bench</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Single Bench">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Single Bench</STRING><STRING></STRING>
<STRING>Single Bench</STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Single Bench</STEXT></LABEL></PART>
</PARTS></CONTROL> <CONTROL ID=79 type="Boolean" name="Multiple Bench">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Multiple Bench</STRING><STRING></STRING>
<STRING>Multiple Bench</STRING><STRING></STRING></STRINGS></MLABEL>
</PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Multiple Bench</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Simple Slope <LF>Short Term">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Simple Slope Short Term</STRING><STRING></STRING>
<STRING>Simple Slope Short Term</STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Simple Slope <LF>Short Term</STEXT></LABEL></PART>
</PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Vertically Sided <LF>Lower Portion">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Vert. Sided Lower Portion</STRING><STRING></STRING>
<STRING>Vert. Sided Lower Portion</STRING><STRING></STRING>
</STRINGS></MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Vertically Sided <LF>
Lower Portion</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Unsupported Vertically<LF>
Sided Lower Portion"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS>
<STRING>Unsupp. Vert. Sided Lower Portion</STRING>
<STRING></STRING><STRING>Unsupp. Vert. Sided Lower Portion</STRING>
<STRING></STRING></STRINGS></MLABEL></PART>

<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Unsupported Vertically<LF>
Sided Lower Portion</STEXT></LABEL></PART></PARTS></CONTROL>
<CONTROL ID=79 type="Boolean" name="Supported Vertically <LF>
Sided Lower Portion"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL><FONT style='B' size=20><STRINGS>
<STRING>Supp. Vert. Sided Lower Portion</STRING><STRING></STRING>
<STRING>Supp. Vert. Sided Lower Portion</STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption">
<LABEL><STEXT><FONT predef=SYSFONT>Supported Vertically <LF>
Sided Lower Portion</STEXT></LABEL></PART></PARTS></CONTROL><LABEL><STEXT>
<FONT style='B' size=24>Based on the type of soil for this project, what <LF>
type of slope can be used for the execavation ?</STEXT></LABEL>
<CONTROL ID=79 type="Boolean" name="Correct Slope Method?">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT size=27><STRINGS><STRING>OFF</STRING>
<STRING>ON</STRING><STRING>OFF</STRING>
<STRING>ON</STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT style='B' size=21>Correct Slope Method?</STEXT></LABEL></PART>
</PARTS></CONTROL><GROUPER><PARTS>
<CONTROL ID=79 type="Boolean" name="Excavation Volume <LF> Choice 1">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING></STRING><STRING></STRING>
<STRING></STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Excavation Volume <LF>
Choice 1</STEXT></LABEL></PART></PARTS></CONTROL>
<CONTROL ID=80 type="Numeric" name="Numeric 1">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT size=16>Numeric 1</STEXT></LABEL></PART>
</PARTS></CONTROL></PARTS></GROUPER><LABEL><STEXT>
<FONT style='B' size=24>Based on the project bid plan, soil type, slope <LF>
method and height:depth ratio, what is the<LF>best approximate volume of
excavation?</STEXT></LABEL>
<CONTROL ID=79 type="Boolean" name="Correct Volume of Excavation?">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT size=25><STRINGS><STRING>OFF</STRING>
<STRING>ON</STRING><STRING>OFF</STRING>
<STRING>ON</STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT style='B' size=21>Correct Volume of Excavation?</STEXT></LABEL></PART>
</PARTS></CONTROL>
<LABEL><STEXT><FONT style='B' size=24>What is the appropriate height:depth ratio based on the
soil type and slope method selected?</STEXT></LABEL>
<CONTROL ID=79 type="Boolean" name="Correct Slope Ratio?">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT size=25><STRINGS><STRING>OFF</STRING>
<STRING>ON</STRING><STRING>OFF</STRING><STRING>ON</STRING>
</STRINGS></MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT style='B' size=21>Correct Slope Ratio?</STEXT></LABEL></PART>
</PARTS></CONTROL><GROUPER><PARTS>
<CONTROL ID=79 type="Boolean" name="Excavation Volume <LF>Choice 2">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text">
<MLABEL><FONT style='B' size=21><STRINGS><STRING></STRING>
<STRING></STRING><STRING></STRING><STRING></STRING>
</STRINGS></MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Excavation Volume <LF> Choice 2</STEXT>
</LABEL></PART></PARTS></CONTROL><CONTROL ID=80 type="Numeric" name="Numeric 2">
<DESC></DESC><TIP></TIP><PARTS><PART ID=82 order=0 type="Caption"><LABEL><STEXT>

<FONT size=16>Numeric 2</STEXT></LABEL></PART>
</PARTS></CONTROL></PARTS></GROUPER><GROUPER><PARTS>
<CONTROL ID=79 type="Boolean" name="Excavation Volume <LF>Choice 4">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING></STRING>
<STRING></STRING><STRING></STRING><STRING></STRING></STRINGS></MLABEL></PART
> <PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Excavation Volume <LF>Choice 4</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=80 type="Numeric" name="Numeric 4">
<DESC></DESC><TIP></TIP><PARTS><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT size=16>Numeric 4</STEXT></LABEL></PART>
</PARTS></CONTROL></PARTS></GROUPER><GROUPER><PARTS>
<CONTROL ID=79 type="Boolean" name="Excavation Volume <LF>Choice 3">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING></STRING><STRING></STRING>
<STRING></STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Excavation Volume <LF>Choice 3</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=80 type="Numeric" name="Numeric 3">
<DESC></DESC><TIP></TIP><PARTS><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT size=16>Numeric 3</STEXT></LABEL></PART>
</PARTS></CONTROL></PARTS></GROUPER>
<CONTROL ID=79 type="Boolean" name="Ratio Choice 2">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>1:1</STRING><STRING></STRING>
<STRING>1:1</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Ratio Choice 2</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Ratio Choice 1">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>3/4 : 1</STRING><STRING></STRING>
<STRING>3/4 : 1</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Ratio Choice 1</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Ratio Choice 3">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>1/2 : 1</STRING><STRING></STRING>
<STRING>1/2 : 1</STRING><STRING></STRING></STRINGS>
</MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT predef=SYSFONT>Ratio Choice 3</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Ratio Choice 4">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=20><STRINGS><STRING>1 1/2 : 1</STRING><STRING></STRING>
<STRING>1 1/2 : 1</STRING><STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Ratio Choice 4</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=91 type="Path" name="UDL file path 6">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>
</STEXT></LABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>UDL file path 6</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="absolute"></PATH></DEFAULT></CONTROL>
<CONTROL ID=91 type="Path" name="UDL file path 7">
<DESC>Path to the Universal Data Link file. </DESC>

<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" ><DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>UDL file path 7</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL>
</PRIV><DEFAULT><PATH type="absolute"></PATH></DEFAULT>
</CONTROL><CONTROL ID=91 type="Path" name="UDL file path 10">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT>
</LABEL></PART><PART ID=82 order=0 type="Caption">
<LABEL><STEXT>UDL file path 10</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="absolute"></PATH></DEFAULT></CONTROL>
<CONTROL ID=91 type="Path" name="UDL file path 8">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>
</STEXT></LABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>UDL file path 8</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT><PATH type="absolute"></PATH>
</DEFAULT></CONTROL><CONTROL ID=79 type="Boolean" name="Rock 3">
<DESC></DESC><TIP></TIP><PARTS>PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING>Flatten Terrain</STRING><STRING></STRING>
<STRING>Flatten Terrain</STRING><STRING></STRING>
</STRINGS></MLABEL></PART><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Rock 3</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="x = y?">
<DESC></DESC><TIP></TIP><PARTS><PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT size=16>x = y?</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=91 type="Path" name="UDL file path 9">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>UDL file path
9</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL>
</PRIV><DEFAULT><PATH type="absolute"></PATH></DEFAULT>
</CONTROL><CONTROL ID=91 type="Path" name="UDL file path 11">
<DESC>Path to the Universal Data Link file. </DESC>
<TIP></TIP><PARTS><PART ID=8019 order=0 type="Graph Plot Legend">
<CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></PART>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL>
<STEXT>UDL file path 11</STEXT></LABEL></PART>

</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT><PATH type="absolute"></PATH>
</DEFAULT></CONTROL><CONTROL ID=91 type="Path" name="UDL file path 12">
<DESC>Path to the Universal Data Link file. </DESC><TIP></TIP><PARTS>
<PART ID=8019 order=0 type="Graph Plot Legend"><CONTROL ID=79 type="Boolean" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</PART><PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>UDL file path
12</STEXT></LABEL></PART></PARTS>
<PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT><PATH type="absolute"></PATH>
</DEFAULT></CONTROL><CONTROL ID=81 type="String" name="data">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART><PART ID=82 order=0 type="Caption">
<LABEL><STEXT><FONT size=16>data</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<CONTROL ID=81 type="String" name="concatenated string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT><FONT style='B' size=24></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT size=16>concatenated
string</STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String" name="data 2"><DESC></DESC><TIP></TIP>
<PARTS><PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT><FONT size=16>data
2</STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String" name="data
3"><DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART><PART ID=82 order=0 type="Caption"><LABEL>
<STEXT><FONT size=16>data 3</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=79 type="Boolean" name="Rock 4">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text">
<MLABEL><FONT style='B' size=21><STRINGS><STRING>Instructions</STRING><STRING>
</STRING><STRING>Instructions</STRING><STRING>
</STRING></STRINGS></MLABEL></PART><PART ID=82 order=0
type="Caption"><LABEL><STEXT><FONT predef=SYSFONT>Rock 4</STEXT></LABEL></PART>
</PARTS></CONTROL><CONTROL ID=79 type="Boolean" name="Rock 5">
<DESC></DESC><TIP></TIP><PARTS><PART ID=22 order=0 type="Boolean Text"><MLABEL>
<FONT style='B' size=21><STRINGS><STRING>Instructions</STRING>
<STRING></STRING><STRING>Instructions</STRING>
<STRING></STRING></STRINGS></MLABEL></PART>
<PART ID=82 order=0 type="Caption"><LABEL><STEXT>
<FONT predef=SYSFONT>Rock 5</STEXT></LABEL></PART>
</PARTS></CONTROL></CONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS><NODE ID=50 type="Global Variable">
<DESC></DESC></NODE><NODE ID=50 type="Global Variable">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="TCP Listen.vi">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><NODE ID=47 type="Function" primID=4E434354 primName="TCP Close
Connection"><DESC></DESC></NODE><NODE ID=140 type="Property Node" name="You Typed:">
<DESC></DESC></NODE><DESC></DESC><TIP></TIP><PARTS></PARTS><NODE ID=44
type="Select"><DESC></DESC><BDCONTENT><NODE ID=99 type="(Un)Bundle by
Name"><DESC></DESC></NODE><CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=87 type="Ring Constant" name="2nd Error to Convert">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=12 order=0 type="Ring Text"><MLABEL><STRINGS><STRING>No Error</STRING>
<STRING>Argument Error</STRING><STRING>Memory Full</STRING>
<STRING>Out of Zone</STRING><STRING>End of File</STRING><STRING>File Already
Open</STRING><STRING>Generic File IO Error</STRING>

<STRING>File Not Found</STRING><STRING>File Permission Error</STRING>
<STRING>Disk Full</STRING><STRING>Duplicate Path</STRING>
<STRING>Too Many Files Open</STRING>
<STRING>Not Enabled</STRING><STRING>Resource File Not Found</STRING>
<STRING>Cannot Add Resource</STRING><STRING>Resource Not Found</STRING>
<STRING>Image Not Found</STRING><STRING>Image Memory Error</STRING>
<STRING>Pen Does Not Exist</STRING><STRING>Config Bad Type</STRING>
<STRING>Config Token Not Found</STRING><STRING>Config Parse Error</STRING>
<STRING>Config Memory Error</STRING><STRING>Bad External Code Format</STRING>
<STRING>Bad External Code Offset</STRING><STRING>External Code Not Present</STRING>
<STRING>Null Window</STRING><STRING>Destroy Window Mixup</STRING>
<STRING>Null Menu</STRING><STRING>Print Aborted</STRING>
<STRING>Bad Print Record</STRING><STRING>Print Driver Error</STRING>
<STRING>Windows Error During Print</STRING><STRING>Memory Error During Print</STRING>
<STRING>Print Dialog Error</STRING><STRING>Geneic Print Error</STRING>
<STRING>Invalid Device Refnum</STRING><STRING>Device Not Found</STRING>
<STRING>Device Parameter Error</STRING><STRING>Device Unit Error</STRING>
<STRING>Can't Open Device</STRING><STRING>Device Call Aborted</STRING>
<STRING>Generic Error</STRING><STRING>Cancelled By User</STRING>
<STRING>Object ID Too Low</STRING><STRING>Object ID Too High</STRING>
<STRING>Object Not In Heap</STRING><STRING>Unknown Heap</STRING>
<STRING>Bad DefProc ID</STRING><STRING>DefProc Not In Table</STRING>
<STRING>Message Out Of Range</STRING><STRING>Null Method</STRING>
<STRING>Unknown Message</STRING><STRING>Manager Call Not Supported</STRING>
<STRING>Bad address</STRING><STRING>Connection In progress</STRING>
<STRING>Connection timed out</STRING><STRING>Connection is already in progress</STRING>
<STRING>Network attribute not supported</STRING><STRING>Network error</STRING>
<STRING>Address in use</STRING><STRING>System out of memory</STRING>
<STRING>Connection aborted</STRING><STRING>Connection refused</STRING>
<STRING>Connection is not connected</STRING><STRING>Connection is already connected</STRING>
<STRING>Connection closed</STRING><STRING>Initialize error (interapplication manager)</STRING>
<STRING>Bad occurrence</STRING><STRING>Wait on unbound occurrence handler</STRING>
<STRING>Occurrence queue overflow</STRING><STRING>Bad path</STRING>
<STRING>Semaphore not signaled</STRING><STRING>Unrecognized type (interapplication manager)</STRING></STRINGS></MLABEL></PART></PARTS></CONTROL>
<CONTROL ID=87 type="Ring Constant" name="3rd Error to Convert">
<DESC></DESC><TIP></TIP><PARTS><PART ID=12 order=0 type="Ring Text">
<MLABEL><STRINGS><STRING>No Error</STRING>
<STRING>Argument Error</STRING><STRING>Memory Full</STRING>
<STRING>Out of Zone</STRING><STRING>End of File</STRING>
<STRING>File Already Open</STRING><STRING>Generic File IO Error</STRING>
<STRING>File Not Found</STRING><STRING>File Permission Error</STRING>
<STRING>Disk Full</STRING><STRING>Duplicate Path</STRING>
<STRING>Too Many Files Open</STRING><STRING>Not Enabled</STRING>
<STRING>Resource File Not Found</STRING><STRING>Cannot Add Resource</STRING>
<STRING>Resource Not Found</STRING><STRING>Image Not Found</STRING>
<STRING>Image Memory Error</STRING><STRING>Pen Does Not Exist</STRING>
<STRING>Config Bad Type</STRING><STRING>Config Token Not Found</STRING>
<STRING>Config Parse Error</STRING><STRING>Config Memory Error</STRING>
<STRING>Bad External Code Format</STRING><STRING>Bad External Code Offset</STRING>
<STRING>External Code Not Present</STRING><STRING>Null Window</STRING>
<STRING>Destroy Window Mixup</STRING><STRING>Null Menu</STRING>
<STRING>Print Aborted</STRING><STRING>Bad Print Record</STRING>
<STRING>Print Driver Error</STRING><STRING>Windows Error During Print</STRING>
<STRING>Memory Error During Print</STRING><STRING>Print Dialog Error</STRING>
<STRING>Generic Print Error</STRING><STRING>Invalid Device Refnum</STRING>
<STRING>Device Not Found</STRING><STRING>Device Parameter Error</STRING>
<STRING>Device Unit Error</STRING><STRING>Can't Open Device</STRING>
<STRING>Device Call Aborted</STRING><STRING>Generic Error</STRING>
<STRING>Cancelled By User</STRING><STRING>Object ID Too Low</STRING>
<STRING>Object ID Too High</STRING><STRING>Object Not In Heap</STRING>
<STRING>Unknown Heap</STRING><STRING>Bad DefProc ID</STRING>

<STRING>DefProc Not In Table</STRING><STRING>Message Out Of Range</STRING>
<STRING>Null Method</STRING><STRING>Unknown Message</STRING>
<STRING>Manager Call Not Supported</STRING>
<STRING>Bad address</STRING><STRING>Connection In progress</STRING>
<STRING>Connection timed out</STRING><STRING>Connection is already in progress</STRING>
<STRING>Network attribute not supported</STRING><STRING>Network error</STRING>
<STRING>Address in use</STRING><STRING>System out of memory</STRING>
<STRING>Connection aborted</STRING><STRING>Connection refused</STRING>
<STRING>Connection is not connected</STRING><STRING>Connection is already connected</STRING>
<STRING>Connection closed</STRING><STRING>Initialize error (interapplication manager)</STRING>
<STRING>Bad occurrence</STRING><STRING>Wait on unbound occurrence handler</STRING>
<STRING>Occurrence queue overflow</STRING><STRING>Bad path</STRING>
<STRING>Semaphore not signaled</STRING><STRING>Unrecognized type (interapplication manager)</STRING></STRINGS></MLABEL></PART></PARTS></CONTROL>
<CONTROL ID=87 type="Ring Constant" name="1st Error to Convert">
<DESC></DESC><TIP></TIP><PARTS><PART ID=12 order=0 type="Ring Text">
<MLABEL><STRINGS><STRING>No Error</STRING>
<STRING>Argument Error</STRING><STRING>Memory Full</STRING>
<STRING>Out of Zone</STRING><STRING>End of File</STRING>
<STRING>File Already Open</STRING><STRING>Generic File IO Error</STRING>
<STRING>File Not Found</STRING><STRING>File Permission Error</STRING>
<STRING>Disk Full</STRING><STRING>Duplicate Path</STRING>
<STRING>Too Many Files Open</STRING><STRING>Not Enabled</STRING>
<STRING>Resource File Not Found</STRING><STRING>Cannot Add Resource</STRING>
<STRING>Resource Not Found</STRING><STRING>Image Not Found</STRING>
<STRING>Image Memory Error</STRING><STRING>Pen Does Not Exist</STRING>
<STRING>Config Bad Type</STRING><STRING>Config Token Not Found</STRING>
<STRING>Config Parse Error</STRING><STRING>Config Memory Error</STRING>
<STRING>Bad External Code Format</STRING><STRING>Bad External Code Offset</STRING>
<STRING>External Code Not Present</STRING><STRING>Null Window</STRING>
<STRING>Destroy Window Mixup</STRING><STRING>Null Menu</STRING>
<STRING>Print Aborted</STRING><STRING>Bad Print Record</STRING>
<STRING>Print Driver Error</STRING><STRING>Windows Error During Print</STRING>
<STRING>Memory Error During Print</STRING><STRING>Print Dialog Error</STRING>
<STRING>Generic Print Error</STRING><STRING>Invalid Device Refnum</STRING>
<STRING>Device Not Found</STRING><STRING>Device Parameter Error</STRING>
<STRING>Device Unit Error</STRING><STRING>Can't Open Device</STRING>
<STRING>Device Call Aborted</STRING><STRING>Generic Error</STRING>
<STRING>Cancelled By User</STRING><STRING>Object ID Too Low</STRING>
<STRING>Object ID Too High</STRING><STRING>Object Not In Heap</STRING>
<STRING>Unknown Heap</STRING><STRING>Bad DefProc ID</STRING>
<STRING>DefProc Not In Table</STRING><STRING>Message Out Of Range</STRING>
<STRING>Null Method</STRING><STRING>Unknown Message</STRING>
<STRING>Manager Call Not Supported</STRING><STRING>Bad address</STRING>
<STRING>Connection In progress</STRING><STRING>Connection timed out</STRING>
<STRING>Connection is already in progress</STRING><STRING>Network attribute not supported</STRING><STRING>Network error</STRING>
<STRING>Address in use</STRING><STRING>System out of memory</STRING>
<STRING>Connection aborted</STRING><STRING>Connection refused</STRING>
<STRING>Connection is not connected</STRING>
<STRING>Connection is already connected</STRING>
<STRING>Connection closed</STRING><STRING>Initialize error (interapplication manager)</STRING>
<STRING>Bad occurrence</STRING><STRING>Wait on unbound occurrence handler</STRING>
<STRING>Occurrence queue overflow</STRING><STRING>Bad path</STRING>
<STRING>Semaphore not signaled</STRING><STRING>Unrecognized type (interapplicationmanager)</STRING></STRINGS></MLABEL></PART></PARTS></CONTROL>
<NODE ID=49 type="SubVI" subVIName="Error to Warning.vi">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="Merge Errors.vi">
<DESC></DESC></NODE><NODE ID=33 type="While Loop" name="Receive loop">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=80 type="Numeric Constant" name="timeout ms (25000)"><DESC></DESC><TIP></TIP><PARTS>

</PARTS></CONTROL><NODE ID=47 type="Function" primID=54574354 primName="TCP Write">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT><NODE ID=202 type="Flat Sequence">
<DESC></DESC><NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS> </PARTS><DEFAULT>
<ARRAY nElems=1><STRING>SoilTypeChosen</STRING>
</ARRAY></DEFAULT><CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>SoilTypeChosen</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT>
<ARRAY nElems=1><STRING>SlopeRatioChosen</STRING>
</ARRAY></DEFAULT><CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=68 type="Index Array">
<DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>Variant To Data returns -2147467261 if the variant
input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0
type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0
type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by
Name"><DESC></DESC></NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives
the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC>
</NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>SelectedSoil</STEXT></LABEL></PART>

</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><CONTROL ID=91 type="Path Constant" name="Application:Directory Path"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT>
</LABEL></PART></PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE><NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi"><DESC></DESC></NODE>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=68 type="Index Array">
<DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>Variant To Data returns -2147467261 if the
variant input is NULL. We want to clear the error and pass back
default data.</STEXT></LABEL></BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS><NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array"><DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT>
</NODE><NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT>
<STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT>	</BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode
gives the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC>
</NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>SelectedSlopeRatio</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">

<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE><NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><CONTROL ID=91 type="Path Constant" name="Application:Directory Path"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database
<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS> </PARTS>
<DEFAULT><ARRAY nElems=1>
<STRING>SlopeMethodChosen</STRING>
</ARRAY></DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>SlopeMethodChosen</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT>
</CONTROL></CONTENT></CONTROL><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=68 type="Index Array"><DESC></DESC></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261
if the variant input is NULL. We want to clear the error
and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP>
<PARTS></PARTS><NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text">
<LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT>
<STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>

<LABEL><STEXT>Make sure the error reported by the XNode
gives the correct source and gives adequate info about the error.
</STEXT></LABEL><NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC>
</NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>SelectedSlopeMethod</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>3/1:1</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>1/2:1</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>1:1</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>1-1/2:1</STRING></DEFAULT>
</CONTROL></BDCONTENT></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP>
<PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>TypeA</STEXT></LABEL></PART></PARTS>
<DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Type B</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Type C</STRING></DEFAULT>
</CONTROL></BDCONTENT></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi"><DESC></DESC>
</NODE><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>

<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the
variant input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT>   <CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>   <PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC></NODE></BDCONTENT>
</NODE></BDCONTENT></NODE></BDCONTENT>
<BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the
XNode gives the correct source and gives adequate info about the error.
</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array">
<DESC></DESC></NODE><NODE ID=68 type="Index Array">
<DESC></DESC></NODE> <CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
<DEFAULT><ARRAY nElems=1><STRING>NamePath</STRING>
</ARRAY></DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>NamePath</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV>
<DEFAULT><PATH type="relative">examples<SEP>database
<SEP>ExcavationDatabase.udl</PATH></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">

```
<DESC></DESC><TIP></TIP>
<PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>IsPossibleToExcavate</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="optional clause">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>WHERE ((PlanNumber = (SELECT RandomValue FROM Random)) AND <LF>
(SoilType = (SELECT SoilTypeChosen FROM SelectedSoil)) AND <LF>
(SlopeRatio = (SELECT SlopeRatioChosen FROM SelectedSlopeRatio)) <LF>
AND (SlopeMethod = (SELECT SlopeMethodChosen FROM
SelectedSlopeMethod)))</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>You wasted money on this excavtion
<LF>because you could have used a 3/4:1 slope</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT>
<STRING>You picked the optimal slope ratio for this excavation which <LF>
is a 3/4:1 ratio.</STRING></DEFAULT></CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP>
<PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS><DEFAULT>
<STRING>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion <LF>
becuse you could have used a 3/4:1 slope</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion <LF>
Becuse you could have used a 3/4:1 slope</STRING></DEFAULT>
</CONTROL></BDCONTENT></NODE></BDCONTENT>
<BDCONTENT><NODE ID=44 type="Select"> <DESC></DESC><BDCONTENT>
<CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text">
<LABEL><STEXT></STEXT></LABEL></PART></PARTS>
```

<DEFAULT><STRING>You picked the optimal slope ratio for this <LF>
excavation which is a 1: 1 ratio.</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion <LF>
becuse you could have used a 1:1 slope ratio</STRING></DEFAULT>
</CONTROL></BDCONTENT></NODE>
</BDCONTENT><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT> <CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STRING></DEFAULT></CONTROL>
</BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>You wasted money on this excavtion as you<LF>
caused damage to the construction site,</STRING></DEFAULT>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" ><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>You picked the optimal slope ratio for this excavation<LF>
which is a slope ratio of 1-1/2:1</STRING></DEFAULT>
</CONTROL></BDCONTENT></NODE></BDCONTENT>
</NODE><CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT>
<ARRAY nElems=1><STRING>Possible</STRING>
</ARRAY></DEFAULT><CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>Possible</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL.
We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP> <PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>

</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT></NODE>
</BDCONTENT><BDCONTENT></BDCONTENT>
</NODE><LABEL><STEXT>Make sure the error reported by the XNode gives the
correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC></NODE><NODE ID=68 type="Index Array">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT>
</LABEL></PART></PARTS><PRIV>
<PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV> <DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><
STEXT>IsPossibleToExcavate</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="optional clause">
<DESC></DESC><TIP></TIP><PARTS> <PART ID=11 order=0 type="Text"><LABEL>
<STEXT>WHERE ((PlanNumber = (SELECT RandomValue FROM Random)) AND <LF>
(SoilType = (SELECT SoilTypeChosen FROM SelectedSoil)) AND <LF>
(SlopeRatio = (SELECT SlopeRatioChosen FROM SelectedSlopeRatio)) <LF>
AND (SlopeMethod = (SELECT SlopeMethodChosen FROM
SelectedSlopeMethod)))</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>Slope Ratio Selected:
</STEXT></LABEL></PART></PARTS>
<DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><DESC></DESC><TIP></TIP>
<PARTS></PARTS><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>Slope Methdo Selected: </STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<CONTROL ID=81 type="String Constant" ><DESC></DESC>
<TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>Simple slope</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">

<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>Soil Type Selected:
</STEXT></LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></BDCONTENT></NODE></NODE>
<CONTROL ID=80 type="Numeric Constant" name="timeout ms (25000)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=54574354 primName="TCP Write">
<DESC></DESC></NODE></BDCONTENT></NODE>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>26</STEXT></LABEL></PART></PARTS>
<DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="timeout ms (25000)">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<CONTROL ID=79 type="False Constant" ><DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=202 type="Flat Sequence">
<DESC></DESC><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>and runs a
query</STEXT></LABEL><CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>DELETE * FROM SelectedSlopeRatio</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Noe">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>
and runs a query</STEXT></LABEL>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>
and runs a query</STEXT></LABEL> <CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>DELETE * FROM SelectedSoil</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>

<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>
and runs a query</STEXT></LABEL>
<CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>DELETE * FROM SelectedSlopeMethod</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>
and runs a query</STEXT></LABEL> <CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>DELETE FROM Random<LF>
WHERE (RandomValue >> -1)</STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT>
</LABEL></PART></PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">

<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20323349 primName="To Long Integer">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE> <LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Random</STRING></DEFAULT>
</CONTROL> <CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT> <NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant
input is NULL. We want to clear the error and pass back default data.</STEXT>
</LABEL></BDCONTENT><BDCONTENT>
<DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC> </NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>

</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings"><DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<CONTROL ID=80 type="Numeric Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC>TIP></TIP>PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<CONTROL ID=80 type="Numeric Constant" name=""><DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is
NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> </BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode
gives the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>Variant To Data returns -2147467261 if the

variant input is NULL. We want to clear the error and pass back default
data.</STEXT></LABEL></BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE> <NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select">
<DESC></DESC> <BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is
NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP>
<PARTS></PARTS><NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE> <NODE ID=32 type="For Loop"><DESC></DESC>
<BDCONTENT>   <CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>   </CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT>
</BDCONTENT></NODE> <LABEL><STEXT>Make sure the error reported by the
XNode gives the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>

<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>Variant To Data returns -2147467261 if the
variant input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP>
<PARTS></PARTS><NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array"><DESC></DESC>
</NODE><NODE ID=32 type="For Loop"><DESC></DESC><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="string"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT>
</NODE><NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
/CONTROL><NODE ID=62 type="Concatenate Strings"><DESC></DESC>
</NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS>   </PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT>
</NODE><LABEL><STEXT>Make sure the error reported by the XNode gives
the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT>   <LABEL><STEXT>Variant To Data returns -2147467261 if the
variant input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP>
<PARTS></PARTS><NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT>   <CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>

</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives
the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE><NODE ID=261 type="XNode">
<DESC></DESC>BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL. We want to clear
the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>

<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL.
We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> </BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT>
</NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL. We want to clear
the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT>

</NODE></BDCONTENT></NODE>
</BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=68 type="Index Array"><DESC></DESC></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL. We want to clear
the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART></PARTS>
<DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT>NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE><NODE ID=68 type="Index Array">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="optional clause">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT>
<STRING>where number = (SELECT RandomValue FROM Random)</STRING></DEFAULT>
</CONTROL><CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT>
<ARRAY nElems=10><STRING>View1</STRING>
<STRING>View2</STRING><STRING>View3</STRING>
<STRING>View4</STRING><STRING>View5</STRING>
<STRING>SoilView</STRING><STRING>Number</STRING>
<STRING>Soil</STRING><STRING>Slope</STRING>
<STRING>Volume</STRING></ARRAY></DEFAULT>
<CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>Soil</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT>

</CONTROL></CONTENT></CONTROL><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>BidPlansAndSoilTypes</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text">
<LABEL><STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Draw Flattened Pixmap.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Read JPEG File.vi">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20503253 primName="String To Path">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC>
<BDCONTENT><NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE>
<LABEL><STEXT>This part connects to a database <LF>and runs a query</STEXT></LABEL>
<CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>DELETE FROM
RandomVolume</STEXT></LABEL></PART> </PARTS>
<DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">

<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>Answer</STEXT>
</LABEL></PART></PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><CONTROL ID=80 type="Numeric Constant" ><DESC></DESC>
<TIP></TIP><PARTS></PARTS></CONTROL><NODE ID=68 type="Index Array">
<DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL.
We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS>/PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings"><DESC></DESC>
</NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE>
<CONTROL ID=81 type="String Constant" name="source"><DESC></DESC>
<TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS> </PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC></NODE>
<CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
<DEFAULT><ARRAY nElems=4><STRING>VolumeAnswer</STRING>
<STRING>VolumeAnswer</STRING><STRING>VolumeAnswer</STRING>
<STRING>VolumeAnswer</STRING></ARRAY>

</DEFAULT><CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>VolumeAnswer</STEXT>
</LABEL></PART></PARTS><DEFAULT><STRING></STRING></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE><NODE ID=33 type="While Loop">
<DESC></DESC><BDCONTENT>
<NODE ID=47 type="Function" primID=4C494543 primName="Round Toward +Infinity">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=2020454C primName="Less Or Equal?">
<DESC></DESC></NODE></BDCONTENT></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<NODE ID=47 type="Function" primID=20343646 primName="To Double Precision Float">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS> </PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">

<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=20343646 primName="To Double Precision Float">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=20343646 primName="To Double Precision Float">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">

<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE>
<CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20343646 primName="To Double Precision Float">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20444441 primName="Add">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=47 type="Function" primID=204C554D primName="Multiply">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=444E4152 primName="Random Number (0-1)">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT>
<NODE ID=47 type="Function" primID=20323349 primName="To Long Integer">
<DESC></DESC></NODE>
<LABEL><STEXT>This part connects to a database <LF>and inserts data</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>RandomVolume</STRING></DEFAULT></CONTROL>
<CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>

<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE>
<NODE ID=140 type="Property Node"><DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE><LABEL><STEXT>This part connects to a database
<LF>and runs a query</STEXT></LABEL>
<CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>DELETE * FROM Answer</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL><NODE ID=52 type="Bundle">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Answer</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS><PART ID=11 order=0 type="Text"><LABEL>
<STEXT>examples\database\ExcavationDatabase.udl</STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node"><DESC></DESC>
</NODE><NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
</NODE></BDCONTENT><BDCONTENT><CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>

<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=202 type="Flat Sequence"><DESC></DESC>
<NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=68 type="Index Array"><DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL.
We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT> <CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"><DESC></DESC>
</NODE><CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT>
<ARRAY nElems=2><STRING>SoilAnswer</STRING>
<STRING>SoilAnswer</STRING></ARRAY> </DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Answer</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">

<DESC></DESC></NODE>
<LABEL><STEXT>This part connects to a database and searches for a column under view 1
<LF>thatmeets the requirements of a query</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>02</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>03</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>04</STRING></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>05</STRING></DEFAULT> </CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>06</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE> <NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT> <CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT><ARRAY nElems=1>
<STRING>SoilTypeChosen</STRING></ARRAY></DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">

<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SelectedSoil</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node"><DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT>
</NODE></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=202 type="Flat Sequence">
<DESC></DESC><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Free Object.vi">
<DESC></DESC></NODE>
<LABEL><STEXT>This part connects to a database <LF>and runs a
query</STEXT></LABEL><CONTROL ID=81 type="String Constant" name="SQL query">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS>
<DEFAULT><STRING>DELETE FROM SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <NODE ID=197 type="PolyVI" subVIName="DB Tools Execute Query.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<LABEL><STEXT>The type of soil depends on which slope method can be used. So we should
get the SoilAnswer and <LF>check to see if the selected slop method is available for that slope before
proceeding.</STEXT></LABEL>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>

</CONTROL><NODE ID=68 type="Index Array">
<DESC></DESC></NODE><NODE ID=261 type="XNode">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><NODE ID=44 type="Select"> <DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array">
<DESC></DESC></NODE><CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
<DEFAULT><ARRAY nElems=2><STRING>SoilAnswer</STRING>
<STRING>SoilAnswer</STRING></ARRAY></DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Answer</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>

<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>This part connects to a database <LF>and inserts data</STEXT></LABEL>
<NODE ID=52 type="Bundle"><DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi"><DESC></DESC>
</NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC> <TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>11</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>08</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>10</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<LABEL><STEXT>This part connects to a database <LF>and inserts data</STEXT></LABEL>

<NODE ID=52 type="Bundle"><DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
DESC></DESC></NODE><NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>09</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>10</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>12</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL><NODE ID=52 type="Bundle"><DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV>

<DEFAULT><PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP> <PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT></CONTROL>
<CONTROL ID=81 type="String Constant" name="y"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>12</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>13</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>14</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL><NODE ID=52 type="Bundle">
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT>
<MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE> <CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT><BDCONTENT><LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL>
<NODE ID=52 type="Bundle"><DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">

<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE> <CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL></BDCONTENT></NODE></BDCONTENT>
</NODE><NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT>
<ARRAY nElems=1><STRING>SlopeMethodChosen</STRING>
</ARRAY></DEFAULT><CONTENT><CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL></CONTENT></CONTROL>
<LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS>
<DEFAULT><STRING>SelectedSlopeMethod</STRING></DEFAULT></CONTROL>
<CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV>
<PROMPT></PROMPT><MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT> <NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
</NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL><CONTROL ID=79 type="False Constant" name="Correct? 4">

<DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL> <NODE ID=202 type="Flat Sequence">
<DESC></DESC><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>This is the soil type</STEXT></LABEL>
<LABEL><STEXT>This is the Slope Method</STEXT></LABEL>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT>
</CONTROL><NODE ID=261 type="XNode"><DESC></DESC>
<BDCONTENT> <NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the
variant input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE> <NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE> <CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></BDCONTENT><BDCONTENT>
</BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE> </BDCONTENT></NODE>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
<NODE ID=68 type="Index Array"><DESC></DESC></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><LABEL><STEXT>Variant To Data returns -2147467261 if the variant
input is NULL. We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC><TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">

<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings"><DESC></DESC>
</NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE>
<CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE></BDCONTENT>
</NODE></BDCONTENT></NODE></BDCONTENT>
<BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=68 type="Index Array"> <DESC></DESC></NODE>
<CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SoilTypeAndMethod</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE> <NODE ID=140 type="Property Node"><DESC></DESC>
</NODE><NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>This is the value sent in from RealXtend</STEXT></LABEL>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC> </NODE><NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT> <CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT><BDCONTENT><CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>16</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT>
</NODE><NODE ID=44 type="Select"><DESC></DESC>
<BDCONTENT><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></BDCONTENT>

<BDCONTENT> <CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>15</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT></NODE>
<CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>11</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>08</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>10</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>10</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT><BDCONTENT> <CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>17</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE>
</BDCONTENT></NODE><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>12</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>08</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>

</PARTS><DEFAULT><STRING>09</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>16</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> </BDCONTENT></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="x"><DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>18</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL></BDCONTENT>
<BDCONTENT> <CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>15</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT>
</NODE><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>12</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>14</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>13</STRING></DEFAULT>
</CONTROL><CONTROL ID=81 type="String Constant" name="y">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>07</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE> <NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS> </PARTS>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" name="Correct? 4">

<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame"><DESC></DESC>
<BDCONTENT><CONTROL ID=82 type="Array Constant" name="columns">
<DESC></DESC><TIP></TIP><PARTS></PARTS><DEFAULT><ARRAY nElems=1>
<STRING>SlopeRatioChosen</STRING></ARRAY></DEFAULT><CONTENT>
<CONTROL ID=81 type="String Constant" name="String">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><SAME_AS_LABEL></DEFAULT></CONTROL>
</CONTENT></CONTROL><LABEL><STEXT>This part connects to a database <LF>
and inserts data</STEXT></LABEL>
<NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE> <CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>SelectedSlopeRatio</STRING></DEFAULT>
</CONTROL><CONTROL ID=79 type="False Constant" name="create table? (f)">
<DESC></DESC><TIP></TIP><PARTS>
</PARTS></CONTROL><CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV>
<PROMPT></PROMPT> <MTCH_PTN></MTCH_PTN><PTN_LBEL></PTN_LBEL>
<STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE>
<NODE ID=49 type="SubVI" subVIName="DB Tools Insert Data.vi">
<DESC></DESC></NODE></BDCONTENT></NODE></NODE>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS> </PARTS></CONTROL>
</BDCONTENT><BDCONTENT><NODE ID=202 type="Flat Sequence">
<DESC></DESC><NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><CONTROL ID=80 type="Numeric Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=68 type="Index Array"><DESC></DESC></NODE>
<NODE ID=261 type="XNode"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>pass through warnings</STEXT></LABEL>
</BDCONTENT><BDCONTENT><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE><NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<LABEL><STEXT>Variant To Data returns -2147467261 if the variant input is NULL.
We want to clear the error and pass back default data.</STEXT></LABEL>
</BDCONTENT><BDCONTENT><DESC></DESC>
<TIP></TIP><PARTS></PARTS>
<NODE ID=47 type="Function" primID=56524431 primName="Reverse 1D Array">
<DESC></DESC></NODE><NODE ID=189 type="Delete From Array">
<DESC></DESC></NODE><NODE ID=32 type="For Loop">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="string">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>->></STRING></DEFAULT>

</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4C4C4143 primName="Call Chain">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="source">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>Database Variant To Data </STRING></DEFAULT>
</CONTROL><NODE ID=62 type="Concatenate Strings">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" name="code">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=99 type="(Un)Bundle by Name"><DESC></DESC>
</NODE></BDCONTENT></NODE></BDCONTENT>
</NODE></BDCONTENT><BDCONTENT></BDCONTENT></NODE>
<LABEL><STEXT>Make sure the error reported by the XNode gives the correct source and
gives adequate info about the error.</STEXT></LABEL>
<NODE ID=47 type="Function" primID=564C4F54 primName="Variant To Data">
<DESC></DESC> </NODE></BDCONTENT></NODE><NODE ID=68 type="Index Array">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" name="table">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>RandomVolume</STRING></DEFAULT>
</CONTROL><NODE ID=49 type="SubVI" subVIName="DB Tools Select Data.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="DB Tools Close Connection.vi">
<DESC></DESC></NODE> <NODE ID=49 type="SubVI" subVIName="Simple Error Handler.vi">
<DESC></DESC></NODE>
<CONTROL ID=91 type="Path Constant" name="Application:Directory Path">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><PRIV><PROMPT></PROMPT><MTCH_PTN></MTCH_PTN>
<PTN_LBEL></PTN_LBEL><STRT_PTH><PATH type="absolute"></PATH></STRT_PTH>
<PTH_BTN_LBL></PTH_BTN_LBL></PRIV><DEFAULT>
<PATH type="relative">examples<SEP>database<SEP>ExcavationDatabase.udl</PATH>
</DEFAULT></CONTROL><NODE ID=47 type="Function" primID=48545042 primName="Build Path">
<DESC></DESC></NODE><NODE ID=140 type="Property Node">
<DESC></DESC></NODE>
<NODE ID=197 type="PolyVI" subVIName="DB Tools Open Connection.vi">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=289 type="Sequence Frame">
<DESC></DESC><BDCONTENT><NODE ID=44 type="Select">
<DESC></DESC><BDCONTENT><CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>19</STRING></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>20</STRING></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT>
<BDCONTENT><CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>21</STRING></DEFAULT></CONTROL>
<NODE ID=47 type="Function" primID=20205145 primName="Equal?">
<DESC></DESC></NODE></BDCONTENT><BDCONTENT>
<CONTROL ID=81 type="String Constant" name="x">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT></STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING>22</STRING></DEFAULT>
</CONTROL><NODE ID=47 type="Function" primID=20205145 primName="Equal?">

<DESC></DESC><NODE></BDCONTENT></NODE>
</BDCONTENT></NODE></NODE>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS>
</CONTROL></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<CONTROL ID=79 type="False Constant" name="Correct? 4">
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT></NODE><CONTROL ID=87 type="Enum" name="mode (standard)">
<DESC></DESC><TIP></TIP><PARTS><PART ID=12 order=0 type="Ring Text">
<MLABEL><STRINGS></STRINGS></MLABEL></PART>
</PARTS></CONTROL><NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><NODE ID=99 type="(Un)Bundle by Name">
<DESC></DESC></NODE> <CONTROL ID=87 type="Ring Constant" name="1st Error to Convert">
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=12 order=0 type="Ring Text"><MLABEL><STRINGS><STRING>No
Error</STRING><STRING>Argument Error</STRING><STRING>Memory
Full</STRING><STRING>Out of Zone</STRING><STRING>End of
File</STRING><STRING>File Already Open</STRING><STRING>Generic File IO
Error</STRING><STRING>File Not Found</STRING><STRING>File Permission
Error</STRING><STRING>Disk Full</STRING><STRING>Duplicate
Path</STRING><STRING>Too Many Files Open</STRING><STRING>Not
Enabled</STRING><STRING>Resource File Not Found</STRING><STRING>Cannot Add
Resource</STRING><STRING>Resource Not Found</STRING><STRING>Image Not
Found</STRING><STRING>Image Memory Error</STRING><STRING>Pen Does Not
Exist</STRING><STRING>Config Bad Type</STRING><STRING>Config Token Not
Found</STRING><STRING>Config Parse Error</STRING><STRING>Config Memory
Error</STRING><STRING>Bad External Code Format</STRING><STRING>Bad External Code
Offset</STRING><STRING>External Code Not Present</STRING><STRING>Null
Window</STRING><STRING>Destroy Window Mixup</STRING><STRING>Null
Menu</STRING><STRING>Print Aborted</STRING><STRING>Bad Print
Record</STRING><STRING>Print Driver Error</STRING><STRING>Windows Error During
Print</STRING><STRING>Memory Error During Print</STRING><STRING>Print Dialog
Error</STRING><STRING>Generic Print Error</STRING><STRING>Invalid Device
Refnum</STRING><STRING>Device Not Found</STRING><STRING>Device Parameter
Error</STRING><STRING>Device Unit Error</STRING><STRING>Can't Open
Device</STRING><STRING>Device Call Aborted</STRING><STRING>Generic
Error</STRING><STRING>Cancelled By User</STRING><STRING>Object ID Too
Low</STRING><STRING>Object ID Too High</STRING><STRING>Object Not In
Heap</STRING><STRING>Unknown Heap</STRING><STRING>Bad DefProc
ID</STRING><STRING>DefProc Not In Table</STRING><STRING>Message Out Of
Range</STRING><STRING>Null Method</STRING><STRING>Unknown
Message</STRING><STRING>Manager Call Not Supported</STRING><STRING>Bad
address</STRING><STRING>Connection In progress</STRING><STRING>Connection timed
out</STRING><STRING>Connection is already in progress</STRING><STRING>Network
attribute not supported</STRING><STRING>Network error</STRING><STRING>Address in
use</STRING><STRING>System out of memory</STRING><STRING>Connection
aborted</STRING><STRING>Connection refused</STRING><STRING>Connection is not
connected</STRING><STRING>Connection is already connected</STRING><STRING>Connection
closed</STRING><STRING>Initialize error (interapplication manager)</STRING><STRING>Bad
occurrence</STRING><STRING>Wait on unbound occurrence
handler</STRING><STRING>Occurrence queue overflow</STRING><STRING>Bad

```
path</STRING><STRING>Semaphore not      signaled</STRING><STRING>Unrecognized type
(interapplication    manager)</STRING></STRINGS></MLABEL></PART>
</PARTS></CONTROL><NODE ID=49 type="SubVI" subVIName="Error to Warning.vi">
<DESC></DESC></NODE><NODE ID=50 type="Global Variable">
<DESC></DESC></NODE><NODE ID=47 type="Function" primID=44524354 primName="TCP
Read"><DESC></DESC>
</NODE><NODE ID=47 type="Function" primID=2020524F primName="Or">
<DESC></DESC></NODE><CONTROL ID=80 type="Numeric Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
<NODE ID=44 type="Select"><DESC></DESC><BDCONTENT>
<CONTROL ID=79 type="False Constant" ><DESC></DESC>
<TIP></TIP><PARTS></PARTS></CONTROL></BDCONTENT><BDCONTENT>
<NODE ID=47 type="Function" primID=4C505324 primName="Search/Split String">
<DESC></DESC></NODE>
<NODE ID=47 type="Function" primID=20304547 primName="Greater Or Equal To 0?">
<DESC></DESC></NODE><CONTROL ID=81 type="String Constant" >
<DESC></DESC><TIP></TIP><PARTS>
<PART ID=11 order=0 type="Text"><LABEL><STEXT>\04</STEXT></LABEL></PART>
</PARTS><DEFAULT><STRING></STRING></DEFAULT></CONTROL>
<NODE ID=62 type="Concatenate Strings"><DESC></DESC>
</NODE></BDCONTENT></NODE>
<NODE ID=47 type="Function" primID=4E454C24 primName="String Length">
<DESC></DESC></NODE></BDCONTENT></NODE>
<NODE ID=50 type="Global Variable">
<DESC></DESC></NODE><NODE ID=50 type="Global Variable">
<DESC></DESC></NODE><DESC></DESC><TIP></TIP><PARTS>
</PARTS></BDCONTENT><BDCONTENT>
<CONTROL ID=79 type="False Constant" >
<DESC></DESC><TIP></TIP><PARTS></PARTS></CONTROL>
</BDCONTENT></NODE></BDCONTENT></VI>
```


Iron Python Code

Below is the code implemented in Iron Python for communicating between the CVRE (LabVIEW) and Simulation Language (LabVIEW)

```
import rxactor
import rxavatar
import sys
import clr
import random
import math
import socket
asm = clr.LoadAssemblyByName('OpenSim.Region.ScriptEngine.Shared')
Vector3 = asm.OpenSim.Region.ScriptEngine.Shared.LSL_Types.Vector3
s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
f=0
class Connection(rxactor.Actor):

    def GetScriptClassName():
        return "LabviewTools.Connection"
    def EventCreated(self):
        global f
        try:
            s.connect(("127.0.0.1",6342))
                except socket.error,msg:
```

```
print "Already connected"
  f = s.makefile('w',1)
  ocketOpen = 1
   super(self.__class__,self).EventCreated()
   print "LabviewTools.Connection EventCreated"
   def EventDestroyed(self):
   print "LabviewTools.Connection EventDestroyed"
  super(self.__class__,self).EventDestroyed()
  def EventTouch(self,vAvatar):
  global f
  print "Button Pressed"
  myid = str(self.llGetObjectName())
  if(myid == "Excavate"):
s.send("26")
f.flush()
print "Receiving message from LabVIEW"
message = s.recv(1024)
print "Coming back with message from LabVIEW"
self.llShout(0, message)
  if(myid == "ConnectToLabVIEWButton"):
                s.send("01")
  f.flush()
  if(myid == "TypeA"):
  s.send("02")
  f.flush()
  if(myid == "TypeB"):
  s.send("03")
  f.flush()
  if(myid == "TypeC"):
  s.send("04")
  f.flush()
  if(myid == "Strata"):
  s.send("05")
  f.flush()
  if(myid == "Rock"):
  s.send("06")
  f.flush()
  if(myid == "SS"):
               s.send("07")
               f.flush()
  if(myid == "SimB"):
               s.send("08")
               f.flush()
  if(myid == "SinB"):
               s.send("09")
               f.flush()
  if(myid == "MB"):
               s.send("10")
               f.flush()
  if(myid == "SSST"):
               s.send("11")
               f.flush()
  if(myid == "VSLP"):
                s.send("12")
               f.flush()
  if(myid == "UVSLP"):
```

```
            s.send("13")
            f.flush()
    if(myid == "SVSLP"):
            s.send("14")
            f.flush()

    if(myid == "ROne"):
            s.send("15")
            f.flush()
    if(myid == "RTwo"):
            s.send("16")
            f.flush()
    if(myid == "RThree"):
            s.send("17")
            f.flush()
    if(myid == "RFour"):
            s.send("18")
            f.flush()
    if(myid == "VolOne"):
            s.send("19")
            f.flush()
    if(myid == "VolTwo"):
            s.send("20")
            f.flush()
    if(myid == "VolThree"):
             s.send("21")
            f.flush()
    if(myid == "VolFour"):
            s.send("22")
            f.flush()
```

Questions on the Concrete Case Study Usability Survey

For the first set of questions students could answer Strongly Agree, Agree, Undecided, Disagree, or Strongly Disagree.

1. Appropriate help was easily accessible whenever additional explanation was requested.
2. The steps required for completion of each task were necessary.
3. There were times when more information or additional tools were needed but unavailable.
4. The instructions for each task were easy to understand.
5. Adequate feedback was provided at the appropriated time.
6. The feedback provided was useful
7. The materials were easily accessible
8. All aspects of the learning interface were clearly labeled.
9. The instructional material was able to be obtained in a timely manner
10. It was easy to refer back to the previously used instructional material.
11. The instructional material was organized and easy to locate.
12. It was sometimes difficult to locate the desired instructional material.
13. The instructions and feedback were consistently arranged.
14. Appropriate instructions on how to proceed were provided when an error was encountered in the environment.
15. Appropriate explanations were provided to minimize errors while achieving the instructional objectives.
16. It was easy to correct errors encountered in the simulation environment.
17. The available tools and instructional material functioned as expected.
18. The instructional material was available at convenient hours of the day.

19. It was easy to stop the learning material and come back later to begin again at the previous stopping point.
20. The user was able to complete a task using more than one approach.
21. All of the instructional material supported the objective of the learning material.
22. The instructional material provided opportunity for decision making.
23. The instructional material was structured in such a way to encourage learning.
24. The materials and activities provide a challenging learning experience.

The next set of questions, student could answer Extremely, Very, Somewhat, Not very or Not at all.

1. How helpful was IN GENERAL the learning material?
2. How helpful was the INSTRUCTINAL OBJECTIVE of the learning material?
3. How helpful was the ENVIRONMENT LOG-IN INSTRUCTIONS section of the learning material?
4. How helpful was the ENVIRONEMENT NAVIGATION INSTRUCTIONS section of the learning material?
5. How helpful was the ENVIRONMENT DOWNLOAD section of the learning material?
6. How helpful was the ENVIRONMENT USER ACCOUNTS section of the learning material?

For the next set of questions students could answer yes or no.

1. Should this activity be accessible to the upcoming students?
2. Should this activity be modified for the upcoming students?

The next questions allowed students to key in their own personal answer.

1. If you thought this activity should be modified for the upcoming students, what modifications would you suggest to make this activity a better educational experience for those students?
2. Comments.

Questions on the Concrete Case Study Motivational Survey

For the first set of questions students could answer Strongly Agree, Agree, Undecided, Disagree, or Strongly Disagree.

1. I had fun doing this activity.
2. I was pretty skilled at this activity.
3. I didn't put too much energy into this activity.
4. I felt pressured while doing this activity.
5. I did this activity because I had no choice.
6. I believe this activity was of some value to me.
7. I put a lot of effort into this activity.
8. I thought this activity was quite enjoyable.
9. I am satisfied with my performance at this task.
10. I think doing this activity will help me better learn the importance and benefits of correct construction excavation.
11. I did the activity because I wanted to.
12. I felt very tense while doing this activity.
13. I tried very hard on this activity.
14. This activity did not hold my attention at all.
15. I was very relaxed in doing this activity.

16. I believe I had some choice about doing this activity.
17. I think this is an important activity.
18. I think that I was pretty good at this activity.
19. While I was doing this activity, I was thinking about how much I enjoyed it.
20. I think this is important to do because it can help me understand the consequences of poor excavation.
21. I thought this was a boring activity
22. I did not feel nervous at all while doing this activity.
23. I felt like I had to do this.
24. I believe doing this activity was beneficial to me.
25. I think I did pretty well at this activity, compared to other students.
26. I was anxious while working on this task
27. I enjoyed doing this activity very much.
28. I didn't really have a choice about doing this task.
29. I would be willing to do this again because it has some value to me.
30. I couldn't do this activity very well.
31. I didn't try very hard to do well at this activity.
32. I think that this activity was useful for learning how to correctly excavate a construction site.
33. I felt pretty competent after working at this activity for a while.
34. I think this activity is very interesting.
35. I felt like it was not my own choice to do this task.
36. It was important to me to do well on this task.

For the next set of questions students could answer yes or no.

1. Should this activity be given to the upcoming students?
2. Should this activity be modified for the upcoming students?

The next questions allowed students to key in their own personal answer.

1. If you answered yes to the previous question, what modifications would you suggest to make to the activity to make it a better educational experience?
2. Comments

Test Case Code

Below is the Simulation Language code implemented in BIG to determine what the visual of the simulation will be in the CVRE (O3D).

Code segment added to BIGs chooseMethodForJob file

```
<iframe id = "coins" src ="http://bigcat.cs.usm.edu/~lduck/cycle7" width="1000" height="500">
        <p>Your browser does not support iframes.</p>
</iframe>

<br><br>

<button type="button" onclick ="foo();"> Submit Answers </button>

<SCRIPT TYPE = "text/javascript">
function foo()
{
        var v = document.getElementsByName("methodForActivity[1]")[0].value;
         var i = document.getElementsByName("methodForActivity[1]")[0].options[v-1].text;
        var s = <%= request.getParameter("jobSize")%>;
```

```
		var p = <%= request.getParameter("jobID")%>;
		var t = <%= request.getParameter("jobType")%>;
		var popup = 
		window.open('http://bigcat.cs.usm.edu/~lduck/cycle7/scripts/write.php?s='+"var jobSize
		= "+s+ "; var jobID =  " + p+ "; var jobType = " + t + ";" + "methodSelected = '" + i + "'
		;", "File wirtten", "menubar=1, resizeable=1, width=350, height=250");
		popup.close();
	}
	</SCRIPT>
```

Communication Code between CVRE and Simulation Language

Below is the php code implemented for communicating between the CVRE (O3D) and Simulation Languge (BIG) in order to complete the simulation for Perdue University.

```
	Write .php
	<?
		$mystring = $_GET['s'];
		$filename = "data.js";
		if(!is_writeable($filename)){
			echo 'The file is not writeable!!';
		}
		$file = fopen($filename, 'w');
		fwrite($file, $mystring);
		fclose($file);
	?>
```

O3D Code to show the visual of the values selected in BIG

Below is the code implemented in the CVRE (O3D) to make the visual based on the calculations of the simulation language (BIG).

```
	$('#PreviewExcavation').click(function () {
		//alert("Want to preview your excavation?");
		var dim_x;
		var dim_z;//objectProperties[2];
		//var scoopCenter = o3djs.math.matrix4.getTranslation(scene.modelsRoot.children[i].localMatrix);
		var st_x;//-scoopCenter[0]-(dim_x/2);
		var st_z;//-scoopCenter[2]-(dim_z/2);
		var dist;//scoopCenter[1] - document.getElementById('defaultDepth').value;
		var slope = 0.75; //default
		var jt;
		var is_done = false;
		var dataSet = $.ajax({
				type: "GET",
				url: "./scripts/data.js",
				dataType: "text",
				async: false,
			}).responseText;

		eval(dataSet.toString());
```

```
		if(jobSize < 20000)
		{
			if (methodSelected == 'Excavation1')
			{
				//worker one
				network.send('{"action":"model", "command":1,
"model":"EE0A569CB7D14996B101B338DE749EFE", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"EE0A569CB7D14996B101B338DE749EFE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

				//worker two
				network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
39.42606702741163,6.661338147750939e-15,1.9014772176742696,1]]}');

				//worker three
				network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

				//worker four
				network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');
			}
			else if (methodSelected == 'Excavation2')
			{
				//worker one
				network.send('{"action":"model", "command":1,
"model":"EE0A569CB7D14996B101B338DE749EFE", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"EE0A569CB7D14996B101B338DE749EFE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

				//worker two
				network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
				network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
39.42606702741163,6.661338147750939e-15,1.9014772176742696,1]]}');

		//worker three
```

```
            network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

            //worker four
            network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
                        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

        //small machine one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //small machine two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');
            //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');
    }
    else if(methodSelected == 'Excavation3')
    {
            //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');
```

```
        //worker two
        network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');
    }
    else if(methodSelected == 'Excavation4')
    {
          //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

        //worker two
        network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('model: size 284: {"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-64.8799046940866,-
1.865174681370263e-14,-34.51404106146414,1]]}');
    }
    else //(methodSelected == 'Subcontractor')
    {
         //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');
```

```
//worker two
network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

//worker three
network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

//worke four
network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

//worker five
network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

//worker six
network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');
//Excavator one
network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

//Excavator two
network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');
```

```
		//Excavator three
		network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258 ", "url":"post/inventory/bulldozer.o3dtgz"}');
		network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

		//Excavator four
		network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
		network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

		//Excavator five
		network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
		network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');


		//Excavator six
		network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
		network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');
	}
		}

		else if ((jobSize >= 20000) && (jobSize < 40000))
		{
				if(methodSelected == 'Excavation1')
				{
			//worker one
						network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
						network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

						//worker two
						network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
						network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-38.97877613096094,-
2.55351295663786e-15,4.145057013386229,1]]}');

						//worker three
```

```
                                        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
                                        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

                                        //worker four
            network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

            //worker five
            network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

            //worker six
            network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
36.15125145976437,1.3322676295501878e-15,13.697831153869629,1]]}');
                              }
                              else if(methodSelected == 'Excavation2')
        {
                              //worker one
            network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

            //worker two
            network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

            //worker three
            network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

          //worke four
          network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');
          //Excavator one
          network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

          //Excavator two
          network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

          //Excavator three
          network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

          //Excavator four
          network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');
      }       else if(methodSelected == 'Excavation3')
      {
           //worker one
          network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

          //worker two
          network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');
        //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

        //Excavator five
        network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-23.2037862158042,1]]}');

        //Excavator six
        network.send('{"action":"model", "command":1, "model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3, "model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]} ');

    }
    else if(methodSelected == 'Excavation4')
    {
            //worker one
        network.send('{"action":"model", "command":1, "model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3, "model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-2.842170943040401e-14,-18.50815467143633,1]]}');

        //worker two
        network.send('{"action":"model", "command":1, "model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3, "model":"4C113BF5BD044D5EAFE8E68B55B44D8A ", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1, "model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3, "model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1, "model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3, "model":"3353647EAD794E7B92C5F4AF5BD9E501", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-60.602827860040314,0,31.18499159753175,1]]} ');

        //worker five
        network.send('{"action":"model", "command":1, "model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3, "model":"37F76728D8C74FBFBAC220F6C50F17DF", "matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
```

```
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

              //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

        //Excavator five
        network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

        //Excavator six
        network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');
    }
```

```
else //(methodSelected == 'Subcontractor')
{
		//worker one
	network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

	//worker two
	network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

	//worker three
	network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

	//worke four
	network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

	//worker five
	network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

	//worker six
	network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');
	//worker seven
	network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

        //worker eight
        network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]} ');

        //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

        //Excavator five
        network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

        //Excavator six
        network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

          //Excavator seven
          network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

          //Excavator eight
          network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');
       }
            }
            else if ((jobSize >=40000) && (jobSize < 80000))
   {
       if(methodSelected == 'Excavation1')
       {
          //worker one
          network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

          //worker two
          network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

          //worker three
          network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

          //worke four
          network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
          network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');
```

```
        //worker five
        network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

        //worker seven
        network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

        //worker eight
        network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]} ');
            //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');
      }
      else if(methodSelected == 'Excavation2')
      {
              //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

        //worker two
        network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

        //worker five
        network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');
        //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

        //Excavator five
        network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

        //Excavator six
        network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

        //Excavator seven
        network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

        //Excavator eight
        network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]} ');
      }
      else if(methodSelected == 'Excavation3')
      {
          //worker one
```

```
	network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

	//worker two
	network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

	//worker three
	network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

	//worke four
	network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]}');

		//Excavator one
	network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

	//Excavator two
	network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

	//Excavator three
	network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');
```

```
	//Excavator four
	network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
	network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

	//Excavator eight
	network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]} ');

	//Excavator nine
	network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

	//Excavator ten
	network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');
```

```
    }
    else if(methodSelected == 'Excavation4')
    {
        //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

        //worker two
        network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

        //worker five
        network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

            //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
```

```
	network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

	//Excavator two
	network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

	//Excavator three
	network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

	//Excavator four
	network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
	network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

	//Excavator eight
```

```
            network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');

            //Excavator nine
            network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

            //Excavator ten
            network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');

            //Excavator eleven
            network.send('{"action":"model", "command":1,
"model":"506A9650552A40729CE19A427C06050A", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"506A9650552A40729CE19A427C06050A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[129.60931526706221,0,-52.4337320168882,1]]}');

            //Excavator twelve
            network.send('{"action":"model", "command":1,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
107.69654083251953,0,22.76034927368164,1]]}');
        }
        else //(methodSelected == 'Subcontractor')
        {
            //worker one
            network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

            //worker two
            network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

        //worker five
        network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

        //worker seven
        network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

        //worker eight
        network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]}');

                //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
```

```
	network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

	//Excavator two
	network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

	//Excavator three
	network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

	//Excavator four
	network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
	network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

	//Excavator eight
```

```
            network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');

            //Excavator nine
            network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

            //Excavator ten
            network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');

            //Excavator eleven
            network.send('{"action":"model", "command":1,
"model":"506A9650552A40729CE19A427C06050A", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"506A9650552A40729CE19A427C06050A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[129.60931526706221,0,-52.4337320168882,1]]}');

            //Excavator twelve
            network.send('{"action":"model", "command":1,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
107.69654083251953,0,22.76034927368164,1]]}');
        }
    }
    else //if (jobSize >= 80000)
    {
        if(methodSelected == 'Excavation1')
        {
            //worker one
            network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

            //worker two
            network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
```

```
	network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

	//worker three
	network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

	//worke four
	network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

	//worker five
	network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

	//worker six
	network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

	//worker seven
	network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

	//worker eight
	network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]}');

		//Excavator one
```

```
            network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

            //Excavator two
            network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

            //Excavator three
            network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

            //Excavator four
            network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

            //Excavator five
            network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

            //Excavator six
            network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');
        }
        else if(methodSelected == 'Excavation2')
        {
            //worker one
            network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

	//worker two
	network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

	//worker three
	network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

	//worke four
	network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

	//worker five
	network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

	//worker six
	network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

	//worker seven
	network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

	//worker eight
	network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
```

```
	network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]} ');

		//Excavator one
	network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

	//Excavator two
	network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

	//Excavator three
	network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

	//Excavator four
	network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
```

```
	network.send('{"action":"model", "command":1, 
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"9BDB608E9FD94DFC81E105E7250F96FB", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

	//Excavator eight
	network.send('{"action":"model", "command":1, 
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"7FE7DDA43725497198A9BED1C478FBEC", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');
	}
	else if(methodSelected == 'Excavation3')
	{
			//worker one
	network.send('{"action":"model", "command":1, 
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

	//worker two
	network.send('{"action":"model", "command":1, 
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

	//worker three
	network.send('{"action":"model", "command":1, 
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

	//worke four
	network.send(' {"action":"model", "command":1, 
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3, 
"model":"3353647EAD794E7B92C5F4AF5BD9E501", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

	//worker five
	network.send('{"action":"model", "command":1, 
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3, 
"model":"37F76728D8C74FBFBAC220F6C50F17DF", 
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
        network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

        //worker seven
        network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

        //worker eight
        network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]}');
        //Excavator one
        network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

        //Excavator two
        network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

        //Excavator three
        network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

        //Excavator four
        network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

        //Excavator five
        network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

        //Excavator six
        network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

        //Excavator seven
        network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

        //Excavator eight
        network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');

        //Excavator nine
        network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

        //Excavator ten
        network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');

        //Excavator eleven
        network.send('{"action":"model", "command":1,
"model":"506A9650552A40729CE19A427C06050A", "url":"post/inventory/bulldozer.o3dtgz"}');
```

```
            network.send('{"action":"model", "command":3,
"model":"506A9650552A40729CE19A427C06050A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[129.60931526706221,0,-52.4337320168882,1]]}');

            //Excavator twelve
            network.send('{"action":"model", "command":1,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660", "url":"post/inventory/bulldozer.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
107.69654083251953,0,22.76034927368164,1]]}');

        }
        else if(methodSelected == 'Excavation4')
        {
            //worker one
            network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

            //worker two
            network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

            //worker three
            network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

            //worke four
            network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
            network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

            //worker five
            network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
            network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');
```

```
//worker six
network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

//worker seven
network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

//worker eight
network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]}');

//Excavator one
network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

//Excavator two
network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

//Excavator three
network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

//Excavator four
network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');

	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
	network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]}');

	//Excavator eight
	network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]}');

	//Excavator nine
	network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

	//Excavator ten
	network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');

	//Excavator eleven
	network.send('{"action":"model", "command":1,
"model":"506A9650552A40729CE19A427C06050A", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"506A9650552A40729CE19A427C06050A",
```

```
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[129.60931526706221,0,-52.4337320168882,1]]}');

        //Excavator twelve
        network.send('{"action":"model", "command":1,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
107.69654083251953,0,22.76034927368164,1]]}');

    }
    else //(methodSelected == 'Subcontractor')
    {
        //worker one
        network.send('{"action":"model", "command":1,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"47C8E83AB1274EB29D1D4C4A54B4BA89",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[135.13965743956678,-
2.842170943040401e-14,-18.50815467143633,1]]}');

        //worker two
        network.send('{"action":"model", "command":1,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"4C113BF5BD044D5EAFE8E68B55B44D8A ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-39.58570955993693,-
1.7763568394002505e-15,-12.634426889900972,1]]}');

        //worker three
        network.send('{"action":"model", "command":1,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"6F0A7CF8C75B4CC78DB21FA9CF200A09",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[131.69121479059834,0,51.65191438765085,1]]}');

        //worke four
        network.send(' {"action":"model", "command":1,
"model":"3353647EAD794E7B92C5F4AF5BD9E501", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
        network.send(' {"action":"model", "command":3,
"model":"3353647EAD794E7B92C5F4AF5BD9E501",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
60.602827860040314,0,31.18499159753175,1]]} ');

        //worker five
        network.send('{"action":"model", "command":1,
"model":"37F76728D8C74FBFBAC220F6C50F17DF", "url":"post/inventory/constructionworker.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"37F76728D8C74FBFBAC220F6C50F17DF",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[128.2470053220489,-
2.842170943040401e-14,31.63657916717301,1]]}');

        //worker six
```

```
	network.send('{"action":"model", "command":1,
"model":"53FFCB946FE143D285FF2990E717A34B", "url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"53FFCB946FE143D285FF2990E717A34B",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
62.15697559559473,0,-16.57000838177396,1]]}');

	//worker seven
	network.send('{"action":"model", "command":1,
"model":"81365DA5E96B47BB8ECA8CD00A10671B",
"url":"post/inventory/constructionworker.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"81365DA5E96B47BB8ECA8CD00A10671B ",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[52.221833928366685,3.1086244689504383e-15,-2.45429200409831,1]]}');

	//worker eight
	network.send(' {"action":"model", "command":1,
"model":"A3B7FDC2E47E4CE480028108558A330D", "url":"post/inventory/wheelbarrowman.o3dtgz"}');
	network.send(' {"action":"model", "command":3,
"model":"A3B7FDC2E47E4CE480028108558A330D",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
86.65466077711334,0,29.83389663696289,1]]}');

	//Excavator one
	network.send('jh read_complete size 118: {"action":"model", "command":1,
"model":"14A50FEBC1444060A4C7C80240434B08", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"14A50FEBC1444060A4C7C80240434B08",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[140.78665634388966,0,-1.78927618885173,1]]}');

	//Excavator two
	network.send('{"action":"model", "command":1,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"AF9C0A05BE9043EC9DC6BE4AA59009C8",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[126.37860834162927,0,25.93783540822082,1]]}');

	//Excavator three
	network.send('{"action":"model", "command":1,
"model":"2B42FEDF2CD04D0DB6527F261A86B258", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2B42FEDF2CD04D0DB6527F261A86B258",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
56.53935193457074,0,14.614388280852424,1]]}');

	//Excavator four
	network.send('{"action":"model", "command":1,
"model":"66712AF341274862BC0819DC2FAA46BC ", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"66712AF341274862BC0819DC2FAA46BC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[160.66843662662248,0,46.8818260648672,1]]}');
```

```
	//Excavator five
	network.send('{"action":"model", "command":1,
"model":"2397559A6AF54AD2889B190B6369DBF0", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"2397559A6AF54AD2889B190B6369DBF0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-47.9664217105905,0,-
23.2037862158042,1]]}');

	//Excavator six
	network.send('{"action":"model", "command":1,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"C6AD686AA0814CC98320AC8BE0B0C2A1",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[127.49722529772939,1.9872992140790302e-14,61.50656352068307,1]]}');

	//Excavator seven
	network.send('{"action":"model", "command":1,
"model":"9BDB608E9FD94DFC81E105E7250F96FB", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"9BDB608E9FD94DFC81E105E7250F96FB",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
98.37848978789316,0,7.239713668823228,1]]} ');

	//Excavator eight
	network.send('{"action":"model", "command":1,
"model":"7FE7DDA43725497198A9BED1C478FBEC", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"7FE7DDA43725497198A9BED1C478FBEC",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[159.1763047017484,0,15.85688695833906,1]]} ');

	//Excavator nine
	network.send('{"action":"model", "command":1,
"model":"436614C9FBCE4432A9A21C8657C85C68", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"436614C9FBCE4432A9A21C8657C85C68",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
95.14780570651371,0,27.215847015380888,1]]}');

	//Excavator ten
	network.send('{"action":"model", "command":1,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"B073E13A0F614CE3955FB9BDAB15D0EE",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
28.27968406677246,0,-60.66792297363281,1]]}');

	//Excavator eleven
	network.send('{"action":"model", "command":1,
"model":"506A9650552A40729CE19A427C06050A", "url":"post/inventory/bulldozer.o3dtgz"}');
	network.send('{"action":"model", "command":3,
"model":"506A9650552A40729CE19A427C06050A",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
```

```
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[129.60931526706221,0,-52.4337320168882,1]]}');

        //Excavator twelve
        network.send('{"action":"model", "command":1,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"C3A950D55CDD4D01BCC3C354C1D3B660",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
107.69654083251953,0,22.76034927368164,1]]}');

        //Excavator thirteen
        network.send('{"action":"model", "command":1,
"model":"673DC1C80062412586BFFB5D6A9830F5", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"673DC1C80062412586BFFB5D6A9830F5",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-10,0],[-
71.98039093454128,0,-29.899430248484123,1]]}');

        //Excavator fourteen
        network.send('{"action":"model", "command":1,
"model":"D723F506ABF64A2889409E04B9C715B0", "url":"post/inventory/bulldozer.o3dtgz"}');
        network.send('{"action":"model", "command":3,
"model":"D723F506ABF64A2889409E04B9C715B0",
"matrix":[[0.029999999329447746,0,0,0],[0,3.5774641582442257e-10,-
0.029999999329447746,0],[0,0.029999999329447746,3.5774641582442257e-
10,0],[184.56648113282162,0,-7.694972006898254,1]]}');
    }
  }
```

APPENDIX B

LETTER OF SUPPORT

CAL POLY

California Polytechnic State University
San Luis Obispo, CA 93407-0284

Construction Management Department
(805) 756-1323 • Fax (805) 756-5740

Hal Johnston
Construction Management Department
California Polytechnic State University
San Luis Obispo, CA 93407-0284
Phone: 805.756.2613 (Office) 805-610-1495 (Cell)
Email: hjohnsto@calpoly.edu

December 16, 2010

Dissertation Committee for Lacey Duckworth
The University of Southern Mississippi
118 College Drive, Box 5106
Hattiesburg, MS 39406

Dear Dissertation Committee for Lacey Duckworth,

Over the last 10 years I and my collaborators have developed the “Building Industry Game” or BIG, a teaching simulation which places students as the managers of construction companies. Students playing BIG make decisions which emulate the thought processes of real construction managers considering price, method, labor, timing capital and so forth. This fills a critical hole in their education by providing hands on experience with the decisions they will make every day as a construction manager. Currently BIG has an entirely web-based text interface. For her dissertation Lacey Duckworth has used BIG as a “proof of concept” to show the effectiveness of her methodology in effectively linking a simulation (such as BIG) with a 3D Visualization in a Collaborative Virtual Reality Environment.

In our most recent meeting Lacey demonstrated a version of BIG with a linked CVRE visualization which was achieved in just a couple of months, provides students with a superior understanding of the decisions they are making and most important seems easily extendible to the entire game with modest effort now that the communication protocol between the two platforms has been established.

I believe that Lacey’s dissertation work is real progress towards richer educational and training materials built upon and extending the capabilities of existing back end simulations such as BIG.

Thanks,

Hal Johnston

Hal Johnston
Professor
Construction Management Department

REFERENCES

Allard, J., Lessage, J. & Raffin, B. (2010 April). Modularity for Large virtual reality applications. *Presence : Teleoperators and Virtual Environments,19(2),* 142-161. doi: 10.1162/pres.19.2.142

Basdogan, C., Ho, C.-H., & Srinivasan, M.A. (2002, August 7). Virtual environments for medical training: graphical and haptic simulation of laparoscopic common bile duct exploration. *IEEE/ASME Transactions on Mechatronics, 6(3),* 269-285. doi: 10.1109/3516.951365

Brana, M. (2003). Virtual reality environment of a building construction timeline. (Master Thesis). Retrieved from http://icee.usm.edu/ICEE/Projects/P-0602-03-A_%28Mark_Brana%29/P-0602-03-A_Home.htm

Brooks, F. P. (1999, November/December). What's real about virtual reality. *IEEE Computer Graphics and Applications*, *19,* 16 - 27. Retrieved from http://www.cs.unc.edu/~brooks/WhatsReal.pdf

Chan, L.K.Y & Lau, H.Y.K. (2004). An interactive virtual reality based e-education system. *Industrial & Manufacturing Systems Engineering: Conference Papers.* doi: 10.1.1.130.1549.pdf

Chang, C., Chang , C. & Chang , S. (2008, October). TMCP: Two-layer multicast communication protocol for bluetooth radio networks. *Computer Networks, 52,* 2764 - 2778. doi: 10.1016/j.comnet2008.6.008

DeWalt. (2005, November 1). *DeWalt construction professional reference*. Pottstown, PA: Pal Publications Inc.

Dorsey, J., Wang, L., Wang, W., Yang, X., Guo, B., & Shum, H-Y. (2006). Real-Time rendering of plant leaves. *Siggraph 2006*, *24,* 712-719. doi: 10.1145/1185657.1185725

Fedkiw, R., Stam, J., & Jensen, H. W. (2001). Visual simulation of smoke. *Computer Graphics SIGGRAPH 01 Proceedings*, 15–22. Retrieved from http://physbam.stanford.edu/~fedkiw/papers/stanford2001-01.pdf

Fisher, J. (2008, Winter). Digging Into Simulation: An SwRI-developed simulator helps train excavator operators safely and effectively. *Technology Today,* 16-19.Retrieved from http://www.swri.org/3pubs/ttoday/Winter08/PDFs/Digging.pdf

Frey, D., Royan, J., Piegay, R., Kermarrec, A.M., Anceaume, E., & Fessant F.L. (2008). Solipsis: a decentralized architecture for virtual environments. *International Workshop on Massively Multiuser Virtual Environments (MMVE'08),* 29-33. doi: 10.1.1.163.2404.pdf

Gehbaeur, F., Zulch, G., Ott, M., & Borkircher, M. (2007, July). Simulation-Based analysis of disturbances in construction operations. *Proceedings IGLC-15.* Retrieved from http://www.lean-management-institut.de/fileadmin/downloads/Lean_Construction/Simulation-Based_Analysis_of_Disturbances_in_Construction_Operations.pdf

Hui, L., Jun, Z., Gong, J., Bingli, X., & Hua, Q. (2010, April). A grid-based collaborative virtual geographic environment for the planning of silt dam systems. *International Journal of Geographical Information Science, 24(4),* 607-621. doi:10.1080/13658810903012425

Leedy, P. D. & Ormrod, J. (2009). *Practical research: planning and design*. Old Tappan, NJ: Prentice Hall

Monahan, T., McArdle, G., & Bertolotto, M. Virtual reality for collaborative e-learning. (May 2008). *Computers and Education, 50 (4)*, 1339-1353. doi:10.1016/j.compedu.2006.12.008

National Institute of Mental Health. (2008, May 7). *Virtual reality, psychotherapy, show promise in treating PTSD symptoms; civilian access to care remains a concern*. Retrieved from http://www.mhasuffolk.org/documents/documents/virtualrealitypsychotherapyPTSD.pdf

realXtend. (2010, February 6). Introduction to realXtend server platform. Retrieved from http://docs.realxtend.org/index.php/Introduction_to_realXtend_Server_Platform

Reynolds, C. (2006). *Crowd simulation on PS3* [PDF document]. Retrieved from http://www.research.scea.com/research/pdfs/GDC2006ReynoldsTemp.pdf

Sampaio, A.Z., Henriques, P.G., & Martins, O.P. (2010). Virtual reality technology used in civil engineering education. (2010). *The Open Virtual Reality Journal, 2,* 18-25.Retrieved from http://www.bentham.org/open/tovrj/articles/V002/18TOVRJ.pdf

SecondLife ™. (2008). *Getting started with LSL*. Retrieved from http://wiki.secondlife.com/wiki/GettingStartedInLSL

Strickland, J. (2007, June 29). *How virtual reality works*. Retrieved from http://electronics.howstuffworks.com/gadgets/other-gadgets/virtual-reality.htm

Wang, P., Ning, P., & Reeves, D. (2007, March). A k-Anonymous communication protocol for overlay networks. *ACM Symposium on Information, Computer and Communication Security.* doi: 10.1145/1229285.1229296

Wikipedia. (2011, February 17). *Simulation*. Retrieved from http://en.wikipedia.org/wiki/Simulation

Wilson, C. (2008, April 9). *Avatars, virtual reality technology, and the U.S. military: emerging policy issues*. Retrieved from http://fas.org/sgp/crs/natsec/RS22857.pdf

Zhu , D., & Tosun, A. (2008, May). *RF communication for LEGO/Handy board with Tmote*. Retireved from http://venom.cs.utsa.edu/dmz/techrep/2008/CS-TR-2008-006.pdf