93 NOV 30 AM 10:54

Integrated Computer-Enhanced Remote Viewing System

RECEIVED

JAN 31 1994

OSTI

Quarterly Report Number 4
for
July - October 1993

November 30, 1993

Work Performed under
Contract No.: DE-AC21-92MC29113

For:
U.S. Department of Energy
Morgantown Energy Technology Center
3610 Collins Ferry Road
Morgantown, West Virginia 26507

By:
Mechanical Technology Incorporated
968 Albany-Shaker Road
Latham, New York 12110
Tel No. (518) 785 - 2800

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED *EB*

1. Introduction

The Interactive, Computer-Enhanced, Remote Viewing System (ICERVS) is a system designed to provide a reliable geometric description of a robotic task space in a fashion that enables robotic remediation to be carried out more efficiently and economically than with present systems. The key elements are a faithful way to store empirical data and a friendly user interface that provides an operator with timely access to all that is known about a scene.

ICERVS will help an operator to analyze a scene and generate additional geometric data for automating significant portions of the remediation activity. Features that enable this include the following:

- Storage and display of empirical sensor data,
- Ability to update segments of the geometric description of the task space,
- Side-by-side comparisons of a live TV scene and a computer generated view of the same scene,
- Ability to create and display computer models of perceived objects in the task space, together with textual comments, and
- Easy export of data to robotic world models for robot guidance.

The development of ICERVS is to occur in three phases.

- Phase 1 will focus on the development of the Data Library, which contains the geometric data about the task space and the objects in it, and the Toolkit, which includes the mechanisms for manipulating and displaying both empirical and model data.
- Phase 2 will concentrate on integrating these subsystems with a sensor subsystem into one working system. Some additional functionality will be incorporated in the Data Library and Toolkit subsystems.
- Phase 3 will expand the configuration to meet the needs of a full scale demonstration of the interactive mapping of some waste site to be identified.

We have just recently begun Phase 2. The Phase 1 activities essentially ended in June and the Phase 2 contract modification was authorized on September 24, 1993. Work on Phase 2 was unofficially begun in August, when the intent to proceed to Phase 2 was clear. This was done in order to meet the schedule of the Buried Waste Integrated Demonstration to be carried out at the Idaho National Engineering Laboratory (INEL) in the fourth quarter of FY94.

In absence of any work on July, this report will cover the technical effort carried out during the quarter consisting of August, September, and October.

2. Summary of Progress to Date

The only Phase 1 activity in the quarter consisted of the modification of the Topical Report covering the activities of Phase 1 (to incorporate the reviewer's comments) and its subsequent release.

The second Phase of the ICERVS project consists of nine tasks which are described in Attachment A. This listing includes the eight tasks originally proposed together with an additional task to supply a working copy of ICERVS to INEL for use with the Buried Waste Integrated Demonstration (BWID).

Significant efforts were devoted to the completion of Task 1: Intermediate System Design, and Task 3: Computer Upgrade. The Intermediate System Design began with a review of the mission requirements for ICERVS, drawing on the User Focus Groups to enhance our understanding of how this system might be used. The Task concluded with the preparation, review, and release of the Phase 2 System Design Report, a copy of which will be found as Attachment B.

The improvements critical to the improved functioning of the computer platform were secured and integrated. These include internal and external memory, a more advanced processor for the Silicon Graphics Indigo workstation and the addition of a Sun Classic workstation to handle the interface to the sensors. The TrueSolid octree software package was acquired for project use and MTI secured several software tools to expedite the development.

3. Progress Report

3.1 Intermediate System Design

The Phase 2 System Design began with a review of three possible missions for the ICERVS. The first involved an underground storage tank as a task space. Others involve buried waste and the decontamination and decommissioning (D&D) of production facilities such as gaseous diffusion plants. Information about such missions was obtained from conversations with the following personnel.

Westinghouse Hanford Co.:
Battelle Pacific Northwest
Laboratory (PNL)

Richard Welty

Dave Bennett
Shawn Bohn

Idaho National Engineering
Laboratory (INEL)

Kevin Kostelnik
Nick Josten

Oak Ridge National Labs :
(ORNL)

Bill Hamel
John Draper

The WHC representative is from Tank Farm Operations; the people from PNL are involved on the development of laser range finders for use in mapping tanks. The INEL personnel are involved in the BWID and the

digface characterization efforts, while the ORNL personnel are concerned with the D&D activities and user interfaces.

As a result of these meetings, our understanding of the potential uses of ICERVS was deepened and the mission profiles of Phase 1 were reviewed and modified. An updated statement of the mission profiles will be found in Appendix B of the Phase 2 System Design Report.

The mission profiles were used to derive the set of functional requirements which the ICERVS must meet. The Phase 1 requirements were updated to reflect the Phase 2 mission profiles and the current statement of these requirements can be found in Appendix D of the Phase 2 System Design Report. A table summarizing the ICERVS functions is included as Table 3-1. This Table also indicates the phased timing for the development and introduction of these features. A comparable listing for the requirements is given in Table 3-2, which is also copied from the Phase 2 System Design Report.

2.1 Sensor Subsystem

The review of the system architecture began with consideration of the sensor subsystem, which is the major addition to the scope of the Phase 1 system. The sensor subsystem represents the source of all input data and, for Phase 2, contains a structured light mapping sensor, a laser range finder mapping sensor, and a collection of sensors that measure physical properties of the material (e.g., temperature, conductivity, radiation level) whose data will be provided to ICERVS through the Minilab subsystem being developed by Sandia National Laboratory (SNL). Architectural data about a plant or factory will also enter ICERVS through the software mechanism associated with the sensor subsystem.

The structured light sensor system is currently under development by MTI as part of a CRADA with both SNL and Oak Ridge National Laboratory (ORNL). A demonstration version of the sensor will be demonstrated at the beginning of February, 1994. This system will be demonstrated operating in a stand-alone mode, but will be capable of remote operation as an ICERVS subsystem. MTI will make this sensor system available to the project to carry out the Phase 2 development.

The laser range finder (LRF) mapping sensor is under study by both ORNL and Pacific Northwest Laboratory (PNL) and a development contract is near completion at Coleman Research Corporation. Discussions with both national laboratories are under way to determine the best way to interface their efforts with ICERVS. Our current goal is to have one of these organizations provide an LRF for the Phase 2 demonstration of ICERVS, but if that cannot be arranged, then to have some real LRF data available for input to ICERVS to show the system's ability to accept, store and display data from a second mapping sensor.

Table 3-1 Phased Implementation Of ICERVS Features

Analysis Software	Computing Platform	Sensor Subsystems
Phase I Features		
<ul style="list-style-type: none"> • Volumetric Data <ul style="list-style-type: none"> - Tree construction and manipulation - Tree structure display - Tree storage display - Tree utility programs • Geometric object data <ul style="list-style-type: none"> - Object definition - Associated text • Display <ul style="list-style-type: none"> - Orthogonal projection - Translation - Scaling - Cut planes along major axes - Multiple windows - Pseudo-color display • Model Building <ul style="list-style-type: none"> - Polygon generator - Region of interest - 3D projection (prismatic models) 	<ul style="list-style-type: none"> • Silicon Graphic workstation (Indigo) • Color monitor • GL graphics library 	<ul style="list-style-type: none"> • Simulated data
Additional Phase II Features		
<ul style="list-style-type: none"> • Volumetric Data <ul style="list-style-type: none"> - Interface for real empirical data - Storage, display and analysis of material property data • Geometric object data <ul style="list-style-type: none"> - Output in IGES format • Display <ul style="list-style-type: none"> - Arbitrary point of view - Arbitrary cut planes - Surface shading - Synchronized video and computer graphics images • Model Building <ul style="list-style-type: none"> - Arbitrary polyhedron - Sculpting of octree model - Quantitative comparison between volumetric data and geometric object 	<ul style="list-style-type: none"> • Real-time CCTV • Operator control (track ball) for interactive viewing • Control interface to sensor subsystem • Upgraded CPU with additional internal memory 	<ul style="list-style-type: none"> • Two subscale instrument stations with <ul style="list-style-type: none"> - Solid state cameras - Laser illuminators - Positioning in pitch • Sensor controller (PC) <ul style="list-style-type: none"> - Image analysis - Surface profiling - Automatic mapping - Interface to computing platform • Simulated waste tank
Additional Phase III Features		
<ul style="list-style-type: none"> • Geometric object data <ul style="list-style-type: none"> - Storage of architectural plans - Input with IGES - Ability to edit IGRIP models • Display with perspective view • Model Building <ul style="list-style-type: none"> - Automatic modeling of simple surfaces - Surface connectivity tool - Surface model templates 	<ul style="list-style-type: none"> • Link to robotic controller (such as COPILOT) 	<ul style="list-style-type: none"> • Three full-scale instrument stations • Color camera and grid projector • Automatic positioning in azimuth • Absolute tank coordinate frame established by surveying tools • Improvement suggested by Phase II • Design of hardened station

Table 3-2 Summary Of ICERVS Requirements

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R1	DATA REPRESENTATION		
R1.01	Octree: spatial data	1	
R1.02	Octree: property data	3	partial for phase 2
R1.03	Octree: spatial interpolation	2	
R1.04	Octree: linear resolution 1:512, expandable	1	
R1.05	Geom: polyhedral objects	1	
R1.06	Geom: geometric primitives	2	
R1.07	Geom: associated text each object	1	
R1.08	Geom: 100 objects, expandable	1	
R1.09	Geom: enter architectural and robot plans	3	design influence
R1.10	Octree: sensor data	3	partial for phase 2
R1.11	Octree: property data interpolation	2	
R2	OBJECT MODELING		
R2.01	Library of primitives / templates	2	
R2.02	Standard templates	2	
R2.03	User-defined templates	3	partial for phase 2
R2.04	Automatic waste surface modeling	3	design influence
R2.05	Synthesize 2D polygons	1	
R2.06	Synthesize 3D polyhedra	2	
R2.07	Dimensioning tools	2	
R2.08	Attach text to geometric objects	1	
R3	COMPUTER GRAPHICS DISPLAY		
R3.01	Translation and scaling	1	
R3.02	Display coordinate axes	2	
R3.03	Parallel cut planes	2	
R3.04	Display geometric object text data	1	
R3.05	Shaded or wire frame geometric object display	2	
R3.06	Update display as points received	1	
R3.07	Pseudo-color octree data	2	
R3.08	Color geometric objects by category	2	
R3.09	Text display view parameters	2	
R3.10	Save / Recall view parameter set	2	
R3.11	Multiple windows displaying same data	1	
R3.12	View tracks sensor station attitude	2	
R3.13	Display 2.5D surface map	2	completed in phase 1
R3.14	Display views of spatial and property data	3	phase 1,2: orthographic phase 2: orthogonal & arbitrary

Table 3-2 Summary Of ICERVS Requirements (cont.)

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R4	VIDEO DISPLAY		
R4.01	Monitor for each camera plus one for processing	3	partial for Phase 2
R4.02	Display wire frame geometric objects over video	3	design influence
R4.03	Real-time CCV	3	phase 2: camera on sensor
R4.04	Flexible color TV camera	3	
R5	MANIPULATION AND ANALYSIS		
R5.01	Copy octree	2	
R5.02	Set region within octree to selected state	2	
R5.03	Operator delete geometric objects	2	
R5.04	Scan object for consistency with octree	2	
R5.05	Compare octree and object data	2	
R5.06	Compare two octrees, compute difference	2	
R5.07	Compute 2.5D surface map from octree	2	
R5.08	Compute difference 2.5D surface maps	2	
R5.09	Surface connectivity	3	design influence
R5.10	Tools for registering data from different sensors	3	partial for phase 2
R5.11	Combine property data via Boolean operations	2	
R6	MISCELLANEOUS FUNCTIONS		
R6.01	Edit system parameters	2	
R6.02	Save / Retrieve waste site data sets to/from disk	1	
R6.03	Build octree from backup raw data	2	
R6.04	Maintain operator log	2	
R6.05	Multiple system of units	2	
R6.06	Define disassembly data	3	design influence
R6.07	Establish Application Data Structure	2	
R6.08	Support multiple active datasets per waste site	2	
R6.09	Support multiple active waste sites	2	
R7	DATA INTERFACE		
R7.01	Input: x,y,z position	2	
R7.02	Input: optional resolution	2	
R7.03	Input: optional property value	2	
R7.04	Input: optional sensor location	2	
R7.05	Input: station angles during visual inspection	2	
R7.06	Output: geometric model data	2	
R7.07	Output: 2.5D surface map data	2	
R7.08	Input: sensor data	3	phase 2: simulate laser range finder and MiniLab

Table 3-2 Summary Of ICERVS Requirements (cont.)

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R8	OPERATOR INTERFACE		
R8.01	Graphic tools	2	
R8.02	Provide operator help function	3	design influence
R8.03	Provide hard copy output	3	design influence
R9	SENSORS		
R9.01	Tele-operate position and rate commands	2	
R9.02	Tele-operation display line of sight	3	partial for phase 2
R9.03	Tele-operation text display station angles	3	partial for phase 2
R9.04	Automatically map surfaces	2	
R9.05	Operator parameters	2	
R9.06	Draw / display scan paths	2	
R9.07	Continual backup of raw data	2	
R9.08	not used		
R9.09	Operate sensor remotely from computing platform	2	
R9.10	Surface mapping sensor	3	partial for phase 2
R9.11	Sensor performance	3	partial for phase 2
R10	SITE ENVIRONMENT		
R10.01	Surface characteristics	3	design influence
R10.02	Illumination and visibility	3	design influence
R10.03	Environmental considerations	3	design influence
R10.04	Design constraints	3	design influence

The Minilab sensor is symbolic of a means to provide to ICERVS the material property data that is needed, e.g., for the buried waste remediation. Discussions with SNL and with the Idaho National Engineering Laboratory (INEL) will determine the details of the interface needed to accept the data used to describe the buried waste pits. Our current goal is to have available sets of experimental data from the BWID program and to demonstrate the ICERVS ability to accept, store and display material property data.

3.3 System Architecture.

The extent of the changes, both extensions for Phase 2 and the modifications of our Phase 1 understandings, indicated the need to rethink the ICERVS system architecture. This was done using the following guidelines.

- Consider the needs of both end users and system integrators of remediation systems.
- Design ICERVS to be consistent with the DOE architecture for remediation systems.
- Develop ICERVS to provide complete and general purpose capabilities.
- Develop ICERVS as a set of independent components.
- Implement ICERVS components as UNIX client-server programs.

The intent and justification of these guidelines is set forth in the Phase 2 System Design Report.

The system design is being done using object oriented techniques. A Phase 2 object oriented analysis (OOA) led to a model which describes the problem that ICERVS is intended to solve. From this, it was determined that the Phase 2 software should be composed of three primary Computer Software Configuration Items (CSCI) :

- Phase 2 Demonstration CSCI
- Volumetric Data CSCI
- Sensor Interface CSCI

and three sensor subsystem CSCIs:

- Structures Light Sensor CSCI
- Laser Range Finder CSCI
- Minilab CSCI

The Volumetric Data CSCI and the three sensor subsystem CSCIs will act as servers, while the Phase 2 Demonstration CSCI will act as a client. The Sensor Interface CSCI will act as both a client and a server. The Phase 2 Demonstration CSCI is the topmost application level process and contains the operator interface. This CSCI may be considered a template for creating a used application.

Each of the CSCIs contains Computer Software Components (CSC) and is described in detail in the attached Phase 2 System Design Report.

3.4 Computer Upgrade

The third project task included the upgrade of the computer platform which MTI has provided to support this project. For Phase 1, the hardware

kernel was a Silicon Graphics (SGI) Indigo workstation with a RISC 3000 CPU, 16 Mbytes of internal RAM, 400 Mbytes of hard disk, 19" color monitor and trackball. The Phase 2 configuration incorporates a RISC 4000 CPU, an additional 16 Mbytes of internal RAM, an additional 540 Mbyte hard disk (for a total system disk space of 940 Mbytes). The color video system will be provided along with the structured light system.

Software upgrades include the TrueSolid software package from Octree Corp., which was purchased for the project. MTI has provided the use of UNIRAS ToolMaster, RogueWave C++ libraries, and several software development tools (debuggers, editors, etc.). From SNL, MTI has obtained a communications software package called ISOE (Intelligent System Operating Environment).

Furthermore, MTI has augmented the SGI Indigo workstation with a Sun SPARC Classic to support the interface to the sensor subsystems. The workstation is configured with a SPARC CPU, 16 Mbytes of internal RAM, 424 Mbytes of hard disk and a 15" color monitor. The software components include UNIX, Open Windows and Motif window managers, the SNL ISOE, and the ORNL structured light stand-alone control code.

In addition to supporting the sensor subsystem, the Sun workstation will provide a networked environment to test ICERVS. It also represents additional resources to facilitate the Phase 2 development.

These computer upgrades were functional by mid September.

4. Plans for Future Activity

The work on Task 2: Sensor Subsystem has begun and is being coordinated with the efforts to complete the CRADA mapping system for its demonstration. The sensor system interface design will continue through this quarter and be completed in January. Effort on Task 4: Analysis Software Design has begun and will be completed by year end, allowing code and test (Task 5) to begin late in December.

5. Assessment of Prospects

The results after the first quarter of activity continue to be quite encouraging. Discussions with end users at various remediation sites reveal ongoing and increasing interest in ICERVS. This Phase will address the needs of the Buried Waste Integrated Demonstration (BWID), while a joint effort with another PRDA project in robotics will begin to explore the utility of ICERVS in the D&D application.

6. Appendices

A. Task Descriptions

B. System Design Report

ATTACHMENT A

ICERVS Task Description

The nine tasks in Phase II are described below.

Task 1: Intermediate System Design

Based on the results from Phase I, MTI will review and update the requirements for the ICERVS, including an evaluation of the "TrueSolid" software and alternative data formats to IGES. The result will be a modified set of requirements for integrating the various subsystems. MTI will complete and initial draft of the ICERVS generic sensor interface. MTI will also establish user focus groups in the DOE end-user community.

Task 2: Sensor Subsystem

MTI will make available a sensor subsystem developed under a CRADA with two national laboratories. This sensor subsystem has two instrument stations. Each station consists of a solid-state camera and a laser projector. Each station will also include the appropriate suspension and positioning hardware for investigating a simulated single shell tank and will provide for yaw and pitch motions.

MTI will upgrade the software in the structured light sensor to operate as an ICERVS sensor subsystem. This will involve defining a remote interface (in accordance with the ICERVS generic sensor interface) and developing the necessary hardware and software for integration with the analysis subsystem. MTI will also define appropriate interfaces for a laser range finder and Minilab. In the absence of empirical sensor data from such sensors, MTI will create simple simulators to test these interfaces.

Task 3: Computer Upgrade

MTI will upgrade the computing platform to enhance the functionality of the overall system. MTI will include, at least: a color TV video monitor, an upgraded central processing unit, and if necessary, additional internal memory. The Octree Corporation's "TrueSolid" software module (or equivalent) to facilitate the Octree engine shall also be included.

Task 4: Analysis Software Design

MTI will review and modify the analysis software requirements as identified in Task 1, Phase II. MTI will design software modules which implement a generic sensor interface, integrate the "TrueSolid" package, and which enhance the capabilities to include, but not be limited to: the ability to accept and store real empirical data (to encompass material properties); the ability to store and merge object models alongside the data they represent; the ability to convert data into a suitable format, such as IGES or STEP, to facilitate interaction with robotic controls. MTI will also design two sets of graphics tools. The first shall enhance the system's display capabilities, and the second shall improve the system's ability to build three-dimensional geometric models.

Task 5: Analysis Software Code and Unit Test

MTI will code, test, and integrate the new analysis software and test it in a stand-alone mode to verify functionality. The new analysis software will incorporate data library functions, interface to "TrueSolid", generic sensor interface, and data output in IGES or an alternative format.

Task 6: System Integration

MTI will integrate, install, and operate data library and toolkit analysis software in an interactive mode to ensure compatibility and interoperability.

MTI will design the software elements needed to ensure effective communication between the sensor subsystems and computing platform. MTI will write, test, and integrate the necessary computer codes required for data acquisition and interactive viewing.

MTI will design an operator interface which provides effective control of the structured light sensor subsystem (i.e. cameras, lights, and positioning systems). MTI will prepare a computer code which allows the operator to control the video camera view selection and simultaneously display a matching computer image.

MTI will interconnect all system components and verify their joint functionality. A simulated, single-shell tank shall also be prepared for demonstrating the subscale ICERVS.

Task 7: Demonstration

MTI will demonstrate and verify, at least, the following components: a sensor subsystem which automatically and interactively maps the surface and walls of a simulated single-shell tank; a sensor interface to a laser range finder; a sensor interface to a minilab subsystem; library analysis software which properly creates and manipulates an equivalent world model in a form compatible with the needs of a robotic controller; analysis software which also demonstrates enhanced viewing and model-building tools; and an ICERVS which demonstrates the ability to match scenes from live TV and the world model as the operator pans the camera.

Task 8: Buried Waste Application

Subtask 8.1: System Design Modifications

MTI will finalize the features and performance needed for the buried waste application through discussions with INEL personnel. MTI will perform design modifications of the ICERVS architecture and algorithms as needed.

Subtask 8.2: Detailed Design

MTI will design the analysis software elements to support the buried waste application, including: enhanced user interface, means for storage and display of

multiple material properties, updating of data from excavation activities, tools for registering data from different sensors, interpolation of material properties for analysis and display, and combination of multiple property data via boolean operations.

Subtask 8.3: Code and Unit Test

MTI will generate the necessary software to implement the features to support the buried waste application. Each element of code shall be unit tested and then tested again at the various levels of integration.

Subtask 8.4: Demonstration, Installation, and Training

MTI will incorporate the buried waste features as part of the ICERVS Phase II demonstration. MTI will install a copy of the ICERVS executable software on an INEL-supplied workstation, provide a user's guide, and supply a one-week level of effort for informal training at INEL.

Task 9: Topical Report and Decision Point

MTI will submit a topical report within sixty (60) days prior to completion of Phase II detailing the work completed to date and including the results of the Task 8 Buried Waste Application. Within thirty (30) days after submittal, the COR shall accept the draft topical report or recommend changes.

Within (3) days after submittal of the topical report, the contracting officer shall decide whether to proceed to the next phase. If the contracting officer decides NOT to continue to the next phase, MTI will submit a camera-ready final copy of the topical report within two weeks of this notification from the contracting officer. This report shall be used in lieu of a final report.

**Integrated Computer Enhanced
Remote Viewing System
(ICERVS)**

**System Design Report
Phase 2**

Prepared for
Department of Energy
Morgantown Energy Technology Center

Prepared under
Contract DE-AC21-92M29113

Prepared by
Mechanical Technology Inc.
968 Albany-Shaker Road
Latham New York 122110
November 30, 1993

Prepared by:	<u>David Smith</u>	<u>11/29/93</u> Date
Approved by:	<u>John Wagner</u>	<u>11/29/93</u> Date

THIS PAGE INTENTIONALLY LEFT BLANK

REVISION HISTORY

<u>Revision</u>	<u>Date</u>	
A	November 15, 1993	Original Version

THIS PAGE INTENTIONALLY LEFT BLANK

Table Of Contents

<u>Description</u>	<u>Page</u>
Cover Page	i
Title Page	ii
Revision History	iii
Table Of Contents	vii
List Of Figures.....	x
List Of Tables	xi
Glossary.....	xii
1. INTRODUCTION.....	1-1
1.1. Scope.....	1-1
1.2. Document Overview	1-1
2. RELATED DOCUMENTS	2-1
3. BACKGROUND.....	3-1
3.1. ICERVS Implementation Phases.....	3-1
3.2. Mission Profiles	3-1
3.3. System Requirements.....	3-3
3.4. Phase I Summary	3-3
3.5. Phase II Success Criteria.....	3-7
4. PHASE II ARCHITECTURE.....	4-1
4.1. System Architecture.....	4-1
4.1.1. Computing Platform	4-1
4.1.2. Sensor Subsystems.....	4-4
4.1.2.1. Structured Light Sensor	4-4
4.1.2.2. Laser Range Finder Sensor Simulator	4-4
4.1.1.3. Sensor MiniLab Simulator	4-6
4.2. Analysis Software Architecture.....	4-6
4.2.1. Guiding Principles.....	4-6
4.2.2. ICERVS Phase II Object Oriented Analysis (OOA) Model.....	4-7
4.2.2.1. What Is The Problem Domain?	4-8
4.2.2.2. What Is OOA?.....	4-8
4.2.2.3. The ICERVS OOA Model	4-8
4.2.3. Phase II Architecture Block Diagram.....	4-10

Table Of Contents (continued)

<u>Description</u>	<u>Page</u>
4.2.4. ICERVS Phase II Demonstration CSCI.....	4-10
4.2.4.1. Top Level User Interface CSC.....	4-10
4.2.4.2. Application Data Structure	4-13
4.2.5. Volumetric Data CSCI.....	4-13
4.2.5.1. Volumetric Data Client Interface CSC	4-15
4.2.5.2. Spatial Data Engine CSC.....	4-15
4.2.5.3. Geometric Object CSC.....	4-15
4.2.5.4. Spatial Data / Object Interactions CSC.....	4-15
4.2.5.5. World Model Interface CSC.....	4-15
4.2.5.6. Property Database CSC.....	4-16
4.2.5.7. Visualization and Interaction CSC.....	4-16
4.2.6. Sensor Interface CSCI.....	4-18
4.2.6.1. Sensor Client Interface CSC	4-18
4.2.6.2. Sensor Subsystem Server Interface CSC.....	4-18
4.2.6.3. Sensor Data Interface CSC.....	4-18
4.2.6.4. Sensors Database CSC.....	4-18
4.2.6.5. Automated Data Acquisition CSC.....	4-20
4.2.6.6. Sensor Data Toolkit CSC	4-20
4.2.7. Sensor Subsystem CSCIs.....	4-20
4.2.7.1. Structured Light Sensor CSCI.....	4-22
4.2.7.2. Laser Range Finder CSCI	4-23
4.2.7.3.. Sensor MiniLab CSCI.....	4-23
4.3. Allocation of System Requirements.....	4-23
 5. COMMON INTERFACE FOR SENSORS.....	 5-1
5.1. General Requirements.....	5-1
5.2. Communications Protocol (ISOE / GENISAS)	5-1
5.3. Applications Program Interface.....	5-2
5.3.1 Standard Commands.....	5-2
5.3.2. Optional Commands	5-2
5.4. Sensor Configuration Requirements.....	5-3
 6. SOFTWARE DESIGN CONSIDERATIONS.....	 6-1
6.1. Object Oriented Methodology	6-1
6.2. Client-Server Communications	6-2
6.3. Purchased Software.....	6-2

Table Of Contents (continued)

<u>Description</u>	<u>Page</u>
7. TESTING	7-1
7.1 Test approach/philosophy	7-1
7.2 CSCI Testing	7-1
7.2.1 Demonstration CSCI Testing	7-1
7.2.2 Volumetric Data CSCI Testing	7-1
7.2.3 Sensor Subsystem CSCI Testing	7-1
7.3 System Testing	7-2
 APPENDIX	
A ICERVS Phase II Mission Profile Revisions.....	A-1
 B Revised Mission Profiles	B-1
B.3.0 Mission Profiles.....	B-1
B.3.1 Underground Storage Tanks	B-1
B.3.1.1 Overview.....	B-1
B.3.1.2 Installation.....	B-1
B.3.1.3 Registration.....	B-2
B.3.1.4 Mapping	B-2
B.3.1.5 Object Modeling.....	B-3
B.3.1.6 Comparison with Previous Map Data	B-4
B.3.1.7 Remediation Data Archival	B-4
B.3.2 Buried Waste Retrieval	B-4
B.3.2.1 Overview.....	B-4
B.3.2.2 Installation.....	B-5
B.3.2.3 Operation	B-5
B.3.3 Production Facility D&D	B-6
B.3.3.1 Overview.....	B-6
B.3.3.2 Operation.....	B-7
 C ICERVS Phase II System Requirements Revisions.....	C-1
C.1. Clarification Of Wording.....	C-1
C.2. Numbering Change	C-4
C.3. New Requirements	C-4
 D Revised System Requirements.....	D-1

List Of Figures

<u>Figure</u>	<u>Description</u>	<u>Page</u>
4-1	ICERVS Functional Model.....	4-2
4-2	ICERVS Phase II System Architecture	4-3
4-3a	MTI Prototype Structured Light Sensor Instrument Stations.....	4-5
4-3b	MTI Prototype Structured Light Sensor Local Control Console	4-5
4-4	ICERVS Phase II OOA Model	4-9
4-5	ICERVS Analysis Software Architecture.....	4-11
4-6a	ICERVS Phase II Top Level Screen and Menu	4-12
4-6b	Volumetric Data Screen	4-12
4-7	Volumetric Data CSCI Block Diagram.....	4-14
4-8a	Volumetric Data Window.....	4-17
4-8b	View Data Window.....	4-17
4-9	Sensor Interface CSCI Block Diagram	4-19
4-10	Generalized Sensor Subsystem CSCI Block Diagram	4-21

List Of Tables

<u>Table</u>	<u>Description</u>	<u>Page</u>
3-1	Phased Implementation Of ICERVS Features.....	3-2
3-2	Summary Of ICERVS Requirements.....	3-4
3-3	Allocation of System Requirements To Phase II Success Criteria.....	3-8
4-1	Allocation Of System Requirements To ICERVS Phase II CSCIs	4-24

THIS PAGE INTENTIONALLY LEFT BLANK

GLOSSARY

2D _

Geometric Object: a geometric object that occupies conceptual space of dimensionality 2. Examples of 2D geometric objects include circles, rectangles, and polygons. A 2D geometric object lies in a plane, occupies surface area, but does not occupy volume.

Polygon: a geometric object characterized by a sequence of connected vertices lying in a plane. The straight line connections between adjacent vertices are called edges. A 2D polygon is *convex* if a line segment joining any two interior points is completely contained within the polygon.

2.5 D Surface Map: projection of a 3D space onto a user specified plane (generally the viewing plane) and is typically represented by a two-dimensional data array, where the two array indices are used to represent two of the dimensional coordinates and the data values are used to represent the third coordinate. The map has a user selectable, uniformly spaced grid.

3 D _

Geometric Object: a geometric object that occupies conceptual space of dimensionality 3. Examples of 3D geometric objects include spheres, rectangular parallelepipeds, and prisms. A 3D geometric object has the property of volume.

Polyhedral Object: a 3D geometric object bounded by plane faces which are polygons.

Swept Volume: the process of creating a 3D geometric object by projecting a 2D geometric object through a path (normal to 2D object) in 3D space.

Category: for a geometric object, the identifier that describes the general type or class for a geometric object, such as, wall, door, barrel, etc.

CCV: closed circuit video. Used to remotely view a location.

Class, C++ _

Base Class: a class which is included in another class(es). A base class typically provides general-purpose characteristics that are tailored by the derived class(es).

Class: a C++ data type defined by the programmer, that aggregates programmer-defined data structures (or member data), member functions, and custom operators.

Derived Class: a class defined such that it includes the member data and member functions from another class. The derived class is said to inherit the characteristics of the other class.

GLOSSARY (continued)

Member Data: the variables and data structures defined within a class (including those inherited from other classes).

Member Function: the member data manipulation functions and other functions for a class (including those inherited from another class).

Client-Server: a style of computer program implementation whereby one program (the server) is configured as a provider of a particular set of services and another program (the client) is a user of those services. The client sends commands to the server which executes the command and returns a result. In a networked environment, clients and servers may reside on different computer systems.

Common Interface for Sensors: A high-level, standard format for communicating sensor commands, data, and user interface information between intelligent sensor subsystems and an application program. The common interface provides a flexible capability for using different sensor configurations in remediation systems.

CRADA: Cooperative Research And Development Agreement signed to transfer technology from a national laboratory to a commercial firm..

CSC: Computer Software Component.

CSCI: Computer Software Configuration Item

CSU: Computer Software Unit

Cut Plane: a plane used to segment 3D space into two regions. Typically one or more cut planes are used to bound a region of interest. A pair of cut planes with a user-setable separation that is used to view slices of the 3D volume is called a slice cut plane.

Derived Property: a property created by the Boolean combination of two or more other properties.

Dialog Box: an area on the user display for user input/output.

DOE: Department Of Energy

GENISAS: General Interface for Supervisor and Subsystems. A software package, developed by Sandia National Laboratory, that provides a set of general communications tools. GENISAS implements a client-server communications model using RPC and UNIX sockets..

GUI: Graphical User Interface.

GLOSSARY (continued)

GISC: Generic Intelligent System Control.

ICERVS: Interactive Computer-Enhanced Remote Viewing System

IGES: Initial Graphics Exchange Standard

IGRIP: A set of software tools developed by Deneb Robotics, Incorporated to facilitate the off-line programming of robotic systems. The tools provide 3D simulation to prototype and visualize robot actions.

INEL: Idaho National Engineering Laboratory

ISOE: Intelligent System Operating Environment. A software package for UNIX low-level, socket-based communications developed by Sandia National Laboratories as part of the GENISAS package.

Level: for a node in a tree, the number of antecedent nodes in the direct path from the node to the root node.

Model _

Geometric Model: a computer representation of a physical object.

World Model: a computer representation, used by robotic systems, of a taskspace and/or workspace.

Motif: An XWindows based computer windowing system from Open Systems Foundation. Motif is a standard part of most UNIX based workstations.

Node: one data element within an octree data structure

Child Node: a direct descendant of another node.

Leaf Node: a node with no children. Leaf nodes of an octree have empty, full, or unknown states.

Node State: for octrees, a flag used to describe the occupancy of the region of space corresponding to a node. The four node states are:

Empty	none of the region is occupied
Partial	some (but not all) of the region is occupied. The occupancy is described in further detail by children nodes

GLOSSARY (continued)

Full all of the region is occupied

Unknown the occupancy of the region is not known. Initially, all nodes are set to the unknown state.

Object_

Category: the identifier that describes the general type or class for a geometric object, such as, wall, door, barrel, etc.

Geometric Object: a representation of a contiguous region of conceptual space defined in a mathematical form(s). Geometric objects can be two dimensional (e.g. circle, rectangle, and polygon) or three dimensional (e.g. sphere, cube, and prism).

Group: an aggregation of two or more contiguous geometric objects that are treated as a single geometric object.

Octree: an 8-ary tree data structure that represents the spatial occupancy of a 3-dimensional region. The data structure is produced by the recursive subdivision of a finite cubical universe.

OOA: object oriented analysis.

OOD: object-oriented design.

OOP: object-oriented programming.

OOPL: object-oriented programming language, such as C++ and SmallTalk.

ORNL: Oak Ridge National Laboratory

Polygon: a geometric object characterized by a sequence of connected vertices lying in a plane. The straight line connections between adjacent vertices are called edges. A 2D polygon is *convex* if a line segment joining any two interior points is completely contained within the polygon.

Polyhedron: a 3D geometric object bounded by plane faces which are polygons

PNL: Pacific Northwest Laboratory

Primitive/Geometric Primitive: a 3D geometric object typically defined by analytic description (equations). Examples of primitives include spheres, cylinders, and planes (half-spaces).

GLOSSARY (continued)

Prismoid: a polyhedron that has all of its vertices in two parallel planes, and with the same number of vertices in each plane.

Property Data: for octrees, data used to describe the non-geometric physical characteristics of a region in space corresponding to one node. Property data is generally a single value (temperature, conductivity, weight, color, etc.) or a simple array of values (magnitude and phase, speed vector, etc.).

Quadtree: a 4-ary tree data structure that represents the spatial occupancy of a 2-dimensional region. The data structure is produced by the recursive subdivision of a finite square universe.

Region: part of a 2D or 3D space that is of interest at a particular point in time.

Remediation: the process of environmental restoration of hazardous sites, such as the waste sites found on DOE reservations.

Robotic System: A computer-controlled mechanical mechanism. Typically, the computer coordinates the actions of multiple linked mechanical elements to achieve a programmed path at the working end of the mechanism.

Rogue Wave Canvas: a software package that provides drawing management for 2D and 3D geometric figures on a XWindows screen.

ROI: Region Of Interest.

Rotation: for computer graphic display, an angular movement of the image about one or more coordinate axes of the computer model of a 3D space.

Scaling: for computer graphic display, a change in the size of the image of a computer-modeled space. For ICERVS, scale is described as a percentage of full scale (At full scale, the entire computer-modeled region is made to exactly fit in the display window.)

Sculpting: for octrees, the process of changing node states to empty along a selected path or within a selected region. Sculpting is typically used to clear regions corresponding to input data provided by line-of-sight sensors.

Sensor -

Laser Range Finder: A non-contact sensor using laser energy to measure range to objects in a scene, which may be one to 20 meters from the sensor.

GLOSSARY (continued)

MiniLab: A computer-based system developed by Sandia National Laboratory for modularly configuring, acquiring data from, and integrating data from a suite of sensors. MiniLab provides a set of common system interfaces and a flexible user interface.

Structured Light: A non-contact, dimensional profiling system that use triangulation between a laser source, projected onto a region of interest, and an imaging detector such as a video camera.

Sensor: a device for measuring some physical quantity, such as temperature, weight, pressure, etc.

Sensor Data: the location, orientation and other attributes of a sensor together with a response set. Typically, sensor data is stored at the node representing the sensor location. One or more kinds of property data can generally be derived from this sensor data.

Sensor Subsystem: a subsystem that contains or simulates at least one sensor and outputs sensor data to another system. A sensor subsystem typically contains a local computer for control and computation. The software on the sensor computing platform is generally configured as a server. Client applications on other computing platforms connect to the sensor subsystem software in order to control the sensor and obtain sensor data.

Site: a particular waste location and its environs. Examples would be an underground storage tank, a buried pit, a production facility, etc.

Slice Cutplane: a pair of cut planes with a user defined separation used to view slices of the 3D volume. The pair of cut planes are controlled as a single object.

SNL: Sandia National Laboratory

Spatial Data: a data base containing discrete elements with dimensionality of three (solids). The data base is used to describe a 3D modeling space.

STEP: Standard for The Exchange of Product model data (ISO 10303). A computer-based method for specifying and exchanging information in engineering and manufacturing systems. STEP defines product specifications, such as geometry/shape, material, tolerances, behavior, function, product structure, and process sequences.

Taskspace: A 3D region of space that is of interest, typically a portion of a workspace. Taskspace and workspace are often used interchangeably.

GLOSSARY (continued)

Translation: for computer graphic display, a change in the apparent position of the image of a computer-modeled space. Translation, when used in conjunction with scaling, provides a means for viewing a subsection of the world model at a magnified scale. For ICERVS, translation is described as screen-horizontal and screen-vertical displacements in external units.

Tree: a data structure characterized by a hierarchy of elements or nodes descendant from a single or root node.

Tree _

Tree Traversal: the process of retrieving each node from a tree in a prescribed order. Typically each node is retrieved one time. A *depth-first* tree traversal retrieves the descendants of a node before retrieving the siblings of a node.

TrueSolid: A software package from Octree Corporation that provides tools for the building, viewing, modification of volumetric data sets using octrees.

Units _

External Units: the physical dimensional units selected by the ICERVS user for display and data input/output.

Internal Units: the physical dimensional units used internally by ICERVS. For ICERVS, these units are SI (dimensions in meters).

Tree Units: the dimensional units used by the octree software to describe octree space. For ICERVS these dimensions vary between zero and 2,097,152 (2²¹). This also sets the maximum spatial resolution of the octree.

UNIRAS Toolmaster: a collect of software products that facilitate development of graphics applications. A 2D and 3D graphics software library (agX) and a GUI builder utility (UIM/X) that facilitates the definition and implementation of GUI components are included in Toolmaster.

UNIX: The computer operating system used on most major workstations. UNIX is a multi-user, multi-tasking computing environment.

Volumetric Data: a data base containing discrete elements with spatial dimensionality of three (solids). The data base is used to describe a 3D modeling space.

VxWorks: A real-time, multi-tasking operating environment for VME-based computers.

GLOSSARY (continued)

Wire frame: a method for defining or generating a graphic display of a polyhedron as a set of points and connecting edges.

Workspace: A 3D region of space that is of interest. A workspace is often subdivided into several taskspaces. In many cases, taskspace and workspace are used interchangeably.

1. INTRODUCTION

1.1. Scope

This ICERVS Phase II System Design Report describes the architecture and preliminary design of the Phase II Interactive Computer-Enhanced Remote Viewing System (ICERVS).

ICERVS is a computer-based system that provides data acquisition, data visualization, data analysis, and model synthesis to support robotic remediation of hazardous environments. Because of the risks associated with hazardous environments, remediation must be conducted remotely using robotic systems, which, in turn, must rely on 3D models of their workspace to support both task and path planning with collision avoidance. Tools such as ICERVS are vital to accomplish remediation tasks in a safe, efficient manner.

The 3D models used by robotic systems are based on solid modeling methods, in which objects are represented by enclosing surfaces (polygons, quadric surfaces, patches, etc.) or collections of primitive solids (cubes, cylinders, etc.). In general, these 3D models must be created and/or verified by actual measurements made in the robotics workspace. However, measurement data is empirical in nature, with typical output being a collection of xyz triplets that represent sample points on some surface(s) in the workspace. As such, empirical data cannot be readily analyzed in terms of geometric representations used in robotic workspace models. The primary objective of ICERVS is to provide a reliable description of a workspace based on dimensional measurement data and to convert that description into 3D models that can be used by robotic systems. ICERVS will thus serve as a critical factor to allow robotic remediation tasks to be performed more effectively (faster, safer) and economically than with present systems.

1.2. Document Overview

Section 1.0 provides an introduction to this document including a definition of scope and an overview of the contents of the document.

Section 2.0 provides a list of related documents that provide ancillary information that may be useful in understanding this document.

Section 3.0 provides a brief background of the ICERVS system including summaries of the implementation phases, mission profiles, system requirements, Phase I results, and the success criteria for Phase II.

Section 4.0 defines the Phase II architecture including the system architecture and the software architecture. An OOA model for ICERVS Phase II is presented. The software CSCIs and CSCs are defined and described. The CSCIs are associated with their respective system requirements.

Section 5.0 contains the ICERVS Common Interface For Sensors Specification. This sensor interface establishes a high-level, standard format for communicating sensor commands, sensor data, and screen display information.

Section 6.0 details several design considerations including object-orient analysis/design methods, the client-server communications approach, and purchased software and software development tools.

Section 7.0 presents a Test Plan for validating the functionality and performance of the Phase II system.

2. RELATED DOCUMENTS

Phase 1 MTI technical proposal Q2-030, "*Interactive Computer Enhanced Remote Viewing System*", December 13, 1991.

Phase 2 MTI technical proposal Q3-419, "*Interactive Computer Enhanced Remote Viewing System*", May 14, 1993.

MTI ICERVS "*Software Development Plan*", December 23, 1992.

MTI "*ICERVS Phase I System Design Report*", December 21, 1992.

MTI "ICERVS Phase 1 Subsystem Design Report", April 27, 1993

MTI 98TR25 "ICERVS Phase 1 Topical Report", August 1993

THIS PAGE INTENTIONALLY LEFT BLANK

3. BACKGROUND

This section provides background information on ICERVS. An overview of the ICERVS system is provided including a discussion of the implementation phases and a review of the mission profiles and system requirements. The results of Phase I are discussed. The final subsection defines the success criteria for Phase II, which provide a framework for the Phase II design and implementation efforts.

3.1. ICERVS Implementation Phases

The development of ICERVS has been structured into three phases based upon the maturity level of its constituent technologies. The features to be developed in each phase are listed in Table 3-1. Each of the ICERVS phases is briefly described below:

The Phase I objective is to achieve Maturity Level III, Subscale Major Subsystems, for those portions of ICERVS that have been demonstrated at the research laboratory scale. The majority of the work in this phase involves the development of the basic analysis software capabilities for octree data storage, sensor data visualization, and geometric object creation and manipulation.

The Phase II objective is to achieve Maturity Level IV, Subscale Integrated System, in which ICERVS will be demonstrated at a limited size to enable successful implementation of a full-scale system in Phase III. The result of this work will be an integrated system that provides a subscale structured light system; an upgraded computing platform and enhanced analysis software to include data analysis and visualization tools; enhanced facilities for creating geometric objects; and an integrated user interface.

The Phase III objective is to achieve Maturity Level V, Full-Scale Demonstration, in which ICERVS will be demonstrated in the intended application. The result of this work will be a full-scale, integrated system for use in the remediation of underground storage tanks. This system will include a full-scale structured light sensor subsystem; an interface to a robot controller; a facility for accepting geometric models derived from architectural data; and enhanced data analysis and visualization tools.

3.2. Mission Profiles

As part of the ICERVS preliminary design, mission profiles were described for three remediation tasks. The original profiles are documented in the ICERVS Phase I System Design Report Section 3. During the initial stages of the Phase II design effort, it became apparent that some minor revisions in the mission profiles were necessary in order to clarify points or to reflect the greater insight into the needs of the DOE community. The details of the revisions to the mission profiles can be found in Appendix A. An updated copy of the Mission Profiles, including the revisions, is contained in Appendix B.

Table 3-1 Phased Implementation Of ICERVS Features

Analysis Software	Computing Platform	Sensor Subsystems
Phase I Features		
<ul style="list-style-type: none"> • Volumetric Data <ul style="list-style-type: none"> - Tree construction and manipulation - Tree structure display - Tree storage display - Tree utility programs • Geometric object data <ul style="list-style-type: none"> - Object definition - Associated text • Display <ul style="list-style-type: none"> - Orthogonal projection - Translation - Scaling - Cut planes along major axes - Multiple windows - Pseudo-color display • Model Building <ul style="list-style-type: none"> - Polygon generator - Region of interest - 3D projection (prismatic models) 	<ul style="list-style-type: none"> • Silicon Graphic workstation (Indigo) • Color monitor • GL graphics library 	<ul style="list-style-type: none"> • Simulated data
Additional Phase II Features		
<ul style="list-style-type: none"> • Volumetric Data <ul style="list-style-type: none"> - Interface for real empirical data - Storage, display and analysis of material property data • Geometric object data <ul style="list-style-type: none"> - Output in IGES format • Display <ul style="list-style-type: none"> - Arbitrary point of view - Arbitrary cut planes - Surface shading - Synchronized video and computer graphics images • Model Building <ul style="list-style-type: none"> - Arbitrary polyhedron - Sculpting of octree model - Quantitative comparison between volumetric data and geometric object 	<ul style="list-style-type: none"> • Real-time CCTV • Operator control (track ball) for interactive viewing • Control interface to sensor subsystem • Upgraded CPU with additional internal memory 	<ul style="list-style-type: none"> • Two subscale instrument stations with <ul style="list-style-type: none"> - Solid state cameras - Laser illuminators - Positioning in pitch • Sensor controller (PC) <ul style="list-style-type: none"> - Image analysis - Surface profiling - Automatic mapping - Interface to computing platform • Simulated waste tank
Additional Phase III Features		
<ul style="list-style-type: none"> • Geometric object data <ul style="list-style-type: none"> - Storage of architectural plans - Input with IGES - Ability to edit IGRIP models • Display with perspective view • Model Building <ul style="list-style-type: none"> - Automatic modeling of simple surfaces - Surface connectivity tool - Surface model templates 	<ul style="list-style-type: none"> • Link to robotic controller (such as COPILOT) 	<ul style="list-style-type: none"> • Three full-scale instrument stations • Color camera and grid projector • Automatic positioning in azimuth • Absolute tank coordinate frame established by surveying tools • Improvement suggested by Phase II • Design of hardened station

3.3. System Requirements

From the mission profiles, a list of ICERVS primary requirements were developed. These requirements are documented in the ICERVS Phase I System Design Report Section 3. In the course of detailing the Phase I subsystem design, it became apparent that some minor revisions in the requirements were necessary in order to clarify points or to reflect the greater insight into the roles of the various subsystem elements. The details of the Phase I revisions to the requirements can be found in Appendix A of the ICERVS Phase I Subsystem Design Report.

During the Phase II design, revisions were made once again to the requirements to reflect the additional insights gained from the Phase I experience. In addition, the modification of the Mission Profiles (paragraph 3.2) and the inclusion of the buried waste task in Phase II has necessitated the addition of several new system requirements. (Table 3-1 was also modified to reflect the changes in the system requirements.) The details of these revisions to the requirements can be found in Appendix C. An updated copy of the ICERVS System Requirements, including all revisions, is contained in Appendix D. The requirements are summarized in Table 3-2 below including annotations to indicate the applicable phase for each requirement.

3.4. Phase I Summary

The Phase I work performed by MTI has brought the ICERVS design to Maturity Level III, Subscale Major Subsystem, and met the established success criteria. This work was performed nearly three months ahead of schedule and within budget. Major Phase I accomplishments include the following:

- 1 As an important first step in the ICERVS design, MTI enumerated and described representative mission profiles for the remediation of underground storage tanks, buried waste sites, and contaminated production facilities. From these profiles, a set of 89 ICERVS system requirements was developed, 35 of which were identified as applicable to Phase I.
- 2 Designed to meet these system requirements, analysis software was implemented in three Computer Software Components (CSCs) related to user interface, octree functions, and object modeling functions, respectively. The software was designed using object-oriented methodology to promote modularity, code reuse, and reduced life-cycle cost. Approximately 50 classes were implemented in C++ source code.
- 3 Testing was conducted to demonstrate that ICERVS met the Phase I success criteria. Data display and object modeling were successfully demonstrated in an integrated system running on an MTI-supplied Silicon Graphics (SGI) workstation. Test results showed that the critical ICERVS elements have achieved Maturity Level III.
- 4 Interest in ICERVS has already been expressed by several DOE end-user sites. INEL believes that ICERVS can be an important addition to buried waste retrieval. Hanford Tank Operations has requested information relative to underground storage tank remediation. ORNL has decided to evaluate ICERVS for potential use relative to robotic decontamination and decommissioning.

Table 3-2 Summary Of ICERVS Requirements

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R1	DATA REPRESENTATION		
R1.01	Octree: spatial data	1	
R1.02	Octree: property data	3	partial for phase 2
R1.03	Octree: spatial interpolation	2	
R1.04	Octree: linear resolution 1:512, expandable	1	
R1.05	Geom: polyhedral objects	1	
R1.06	Geom: geometric primitives	2	
R1.07	Geom: associated text each object	1	
R1.08	Geom: 100 objects, expandable	1	
R1.09	Geom: enter architectural and robot plans	3	design influence
R1.10	Octree: sensor data	3	partial for phase 2
R1.11	Octree: property data interpolation	2	
R2	OBJECT MODELING		
R2.01	Library of primitives / templates	2	
R2.02	Standard templates	2	
R2.03	User-defined templates	3	partial for phase 2
R2.04	Automatic waste surface modeling	3	design influence
R2.05	Synthesize 2D polygons	1	
R2.06	Synthesize 3D polyhedra	2	
R2.07	Dimensioning tools	2	
R2.08	Attach text to geometric objects	1	
R3	COMPUTER GRAPHICS DISPLAY		
R3.01	Translation and scaling	1	
R3.02	Display coordinate axes	2	
R3.03	Parallel cut planes	2	
R3.04	Display geometric object text data	1	
R3.05	Shaded or wire frame geometric object display	2	
R3.06	Update display as points received	1	
R3.07	Pseudo-color octree data	2	
R3.08	Color geometric objects by category	2	
R3.09	Text display view parameters	2	
R3.10	Save / Recall view parameter set	2	
R3.11	Multiple windows displaying same data	1	
R3.12	View tracks sensor station attitude	2	
R3.13	Display 2.5D surface map	2	completed in phase 1
R3.14	Display views of spatial and property data	3	phase 1,2: orthographic phase 2: orthogonal & arbitrary

Table 3-2 Summary Of ICERVS Requirements (cont.)

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R4	VIDEO DISPLAY		
R4.01	Monitor for each camera plus one for processing	3	partial for Phase 2
R4.02	Display wire frame geometric objects over video	3	design influence
R4.03	Real-time CCV	3	phase 2: camera on sensor
R4.04	Flexible color TV camera	3	
R5	MANIPULATION AND ANALYSIS		
R5.01	Copy octree	2	
R5.02	Set region within octree to selected state	2	
R5.03	Operator delete geometric objects	2	
R5.04	Scan object for consistency with octree	2	
R5.05	Compare octree and object data	2	
R5.06	Compare two octrees, compute difference	2	
R5.07	Compute 2.5D surface map from octree	2	
R5.08	Compute difference 2.5D surface maps	2	
R5.09	Surface connectivity	3	design influence
R5.10	Tools for registering data from different sensors	3	partial for phase 2
R5.11	Combine property data via Boolean operations	2	
R6	MISCELLANEOUS FUNCTIONS		
R6.01	Edit system parameters	2	
R6.02	Save / Retrieve waste site data sets to/from disk	1	
R6.03	Build octree from backup raw data	2	
R6.04	Maintain operator log	2	
R6.05	Multiple system of units	2	
R6.06	Define disassembly data	3	design influence
R6.07	Establish Application Data Structure	2	
R6.08	Support multiple active datasets per waste site	2	
R6.09	Support multiple active waste sites	2	
R7	DATA INTERFACE		
R7.01	Input: x,y,z position	2	
R7.02	Input: optional resolution	2	
R7.03	Input: optional property value	2	
R7.04	Input: optional sensor location	2	
R7.05	Input: station angles during visual inspection	2	
R7.06	Output: geometric model data	2	
R7.07	Output: 2.5D surface map data	2	
R7.08	Input: sensor data	3	phase 2: simulate laser range finder and MiniLab

Table 3-2 Summary Of ICERVS Requirements (cont')

NO	SYSTEM REQUIREMENT	PHASE	NOTES
R8	OPERATOR INTERFACE		
R8.01	Graphic tools	2	
R8.02	Provide operator help function	3	design influence
R8.03	Provide hard copy output	3	design influence
R9	SENSORS		
R9.01	Tele-operate position and rate commands	2	
R9.02	Tele-operation display line of sight	3	partial for phase 2
R9.03	Tele-operation text display station angles	3	partial for phase 2
R9.04	Automatically map surfaces	2	
R9.05	Operator parameters	2	
R9.06	Draw / display scan paths	2	
R9.07	Continual backup of raw data	2	
R9.08	not used		
R9.09	Operate sensor remotely from computing platform	2	
R9.10	Surface mapping sensor	3	partial for phase 2
R9.11	Sensor performance	3	partial for phase 2
R10	SITE ENVIRONMENT		
R10.01	Surface characteristics	3	design influence
R10.02	Illumination and visibility	3	design influence
R10.03	Environmental considerations	3	design influence
R10.04	Design constraints	3	design influence

3.5. Phase II Success Criteria

This program aspires to make ICERVS the de facto standard for volumetric data storage, manipulation and presentation in environmental restoration systems. Toward that goal, the ICERVS Phase II system will be successful if:

- 1 The sensor subsystem automatically and/or interactively maps the surface and walls of the simulated single-shell tank.
- 2 The ICERVS properly creates a volumetric data base for the simulated single-shell tank and maintains an equivalent world model in a form compatible with the needs of robotic controllers.
- 3 The ICERVS demonstrates enhanced viewing and model-building tools.
- 4 The ICERVS displays corresponding (matching) scenes from live CCV and the computer's world model as the operator pans the camera.

The specific success criteria for the buried waste material properties task with INEL are:

- 5 Accept, align, store, and display material property data from two or more sensors that have scanned a selected segment of the workspace.
- 6 Fill in missing material data points through interpolation.
- 7 Combine two sets of property data by Boolean means to form a third (derived) property, store, and display it in an appropriate manner.
- 8 Update the volumetric and geometric object data, as needed, in response to partial excavation.

Table 3-3 relates the success criteria and the system requirements. In some cases, a single requirement may be supportive of several success criteria, in which case the requirement will be listed multiple times.

Table 3-3 Allocation Of System Requirements To Phase II Success Criteria

No.	Success Criteria	Full Implementation	Partial Implementation
1	Automatically or interactively maps surface and walls of simulated tank	R9.01, R9.04, R9.05, R9.06, R9.07, R9.09	R9.02, R9.03, R9.10, R9.11
2	Properly creates volumetric data base and maintains equivalent world model.	R1.01, R1.03, R1.04, R1.05, R1.06, R1.07, R1.08, R5.01, R5.03, R5.04, R5.05, R5.06, R5.07, R5.08, R6.01, R6.02, R6.03, R6.04, R6.05, R6.07, R6.08, R6.09, R7.01, R7.02, R7.06, R7.07	
3	Demonstrates enhanced 1. model building tools and 2. viewing tools	R2.01, R2.02, R2.05, R2.06, R2.07, R2.08, R3.04, R3.05, R3.08 R3.01, R3.02, R3.03, R3.06, R3.07, R3.09, R3.10, R3.11, R3.12, R3.13, R8.01	R2.03 R3.14
4	Displays corresponding scenes from live CCV and world model.	R3.12, R7.04, R7.05, R9.01, R9.09	R4.01, R4.03, R9.02, R9.03
5	Accept, align, store, and display sensor property data	R7.03, R9.09	R1.02, R1.10, R3.14, R5.10, R7.08
6	Interpolate sparse material property data	R1.11	
7	Combine two sets of property data by Boolean means.	R5.11	R3.14
8	Update the geometric data in response to partial excavation.	R5.01, R5.02, R5.04, R5.06	

4. PHASE II ARCHITECTURE

In reviewing the various remediation missions, it becomes evident that ICERVS can be used in missions that possess differing data requirements and therefore use differing sensor equipment, processing algorithms, and data stores. To address these differing remediation environments and needs, ICERVS Phase II is being based upon a flexible architecture.

4.1. System Architecture

In general, an ICERVS will consist of a computing platform, one or more sensor subsystems for data acquisition, one or more CCV subsystems, an interface to a robotic controller/simulator, and analysis software for data processing. The system outputs include world model information, CAD models, 2.5D maps, and data for archiving. Figure 4-1 summarizes the ICERVS system and illustrates its functional model. Figure 4-2 shows the ICERVS architecture that will be demonstrated as part of Phase II. The paragraphs that follow describe the Phase II system architecture including the computing platform and the sensor subsystems. The next section (4.2) describes the ICERVS software architecture.

4.1.1. Computing Platform

The hardware kernel of ICERVS is a Silicon Graphics (SGI) Indigo workstation. The Phase I Indigo configuration included a RISC 3000 CPU, 16 Mbyte memory, 400 Mbyte hard disk, 19" color monitor, and trackball. For Phase II, the Indigo is being upgraded to incorporate a RISC 4000 CPU, an additional 16 Mbyte memory (32 Mbyte total), and an additional 540 Mbyte hard disk (940 Mbyte total). SGI system software upgrades include UNIRAS ToolMaster, Rogue Wave C++ library updates, SNL ISOE (Intelligent System Operating Environment), Octree Corporation's TrueSolid software, and several software development tools (debuggers, editors, etc.).

A Sun SPARC Classic workstation has been added to augment the ICERVS SGI computing platform. The Sun workstation is configured with a SPARC CPU, 16 Mbyte memory, 424 Mbyte hard disk, and a 15" color monitor. Major system software components include UNIX, Open Windows and Motif window managers, SNL ISOE, and the ORNL structured light standalone control code. The main functions of the Sun for the ICERVS Phase II effort are:

1. Provide additional software development resources to facilitate the implementation of Phase II
2. Provide a networked environment to test ICERVS
3. Support the ICERVS interface with the structured light sensor subsystem, the laser range finder simulator, and the MiniLab simulator.

Network communication is by means of Ethernet. All computing elements in an ICERVS system are assumed to be connected to the network. All of the above hardware and software have been supplied by MTI, with the exception of the ISOE and structured light standalone control code which have been provided by the National Labs.

Figure 4-1 -- ICERVS General System Architecture

Figure 4-2 ICERVS Phase II System Architecture

4.1.2. Sensor Subsystems

The data acquisition part of ICERVS Phase II is configurable in order to meet the needs of the specific application. For example, dimensional measurements can be acquired by structured light and/or laser range sensors. Material property measurements can be acquired by the various sensors found in the DOE MiniLab system or by off-line means such as core sampling. Each sensor subsystem will typically include a local computer for control and computation. As such, each sensor subsystem can be operated as a stand-alone system if desired.

However, key to each sensor subsystem is the capability of operating remotely from an integrated user interface resident on the ICERVS computing platform. To support different sensor configurations, the ICERVS system design includes a Common Interface for Sensors for communications between the computing platform and the sensor subsystems. This interface will establish a high-level, standard format for communicating sensor commands, sensor data, and screen display information. Each sensor subsystem will define and implement their individual commands, data, and screen displays to be consistent with the standard. In some cases, multiple sensor subsystem may be implemented on a single computing platform.

For ICERVS Phase II, the three sensor subsystems (Structured Light Sensor, Laser Range Finder Simulator, and MiniLab Simulator) will all be implemented on the Sun Classic platform. The software for each sensor subsystem will act as a server and will interface to the SGI-based computing platform analysis software package. Each of the Phase II sensor subsystems is briefly described below.

4.1.2.1. Structured Light Sensor

As part of CRADAs with Sandia and Oak Ridge, MTI is developing a structured light sensor subsystem. An early prototype of the system consisting of two instrument stations and a 486/DOS personal computer is shown in Figure 4-3. One instrument station projects a plane of laser light onto the measurement surface. The second instrument station uses a high-resolution video camera to image the resulting laser line. Both instrument stations include motors for rotary positioning. Application software provides for the control of the instrument stations, processing of image data, calculation of 3D measurement results, presentation of results, and user interface functions.

A four inch (4") diameter, single station version of the sensor is presently being developed under the CRADAs. During ICERVS Phase II, this structured light sensor and its controller will be integrated with the Sun Classic workstation that will then interface to the ICERVS computing platform.

4.1.2.2. Laser Range Finder Sensor Simulator

At Pacific Northwest Laboratory (PNL), DOE is investigating laser range finder sensors as possible mapping sensors for environmental remediation. ICERVS Phase II will simulate this sensor on the Sun Classic. MTI will design a remotely interfaced CSCI in accordance with the ICERVS Common Interface For Sensors standard. If possible, actual measurement data from PNL will be incorporated into the simulator; otherwise, a mathematical model will be used to generate appropriate data.

V93-132

Figure 4-3a -- MTI Prototype Structured Light Sensor Instrument Stations

V93-130

Figure 4-3b -- MTI Prototype Structured Light Sensor Local Control Console

4.1.1.3. Sensor MiniLab Simulator

At Sandia National Laboratory (SNL), DOE is developing a MiniLab to acquire sensor data on material properties of various waste sites. ICERVS Phase II will simulate MiniLab on the Sun Classic. MTI will design a remotely interfaced CSCI in accordance with the ICERVS Common Interface For Sensors standard. If possible, actual measurement data from Sandia and/or INEL will be incorporated into the simulator; otherwise, a mathematical model will be used to generate data for material properties, such as temperature or electrical conductivity.

4.2. Analysis Software Architecture

The ICERVS analysis software runs on the SGI computing platform, interfaces with one or more sensor subsystems, converts sensor data into a common data structure, and stores such data from multiple sensor subsystems. The data from the sensors can be stored together with spatial information to generate a volumetric description of the workspace. The analysis software can provide a visual display of the volumetric and property data for a workspace from arbitrary viewpoints, with numerous tools for modifying, selecting, or highlighting the data according to its characteristics and/or regions of interest. These capabilities allow an operator to easily analyze data to obtain maximum insight prior to and during robotic operations.

Also included in the ICERVS analysis software is a geometric modeling capability that enables the operator to interactively define, manipulate, and output geometric objects that represent features of interest in the workspace. In waste remediation applications, these features would include barrels, pipes, support structures, instrumentation, or solidified waste. By defining models for these features and passing them to the robotic system, the remediation tasks can be more readily accomplished with an accurate 3D description of the "as-is" condition of the workspace.

The analysis software architecture for Phase I was fairly simple; it consisted of a single CSCI (IcerMain) containing three CSCs (Octree Engine, Object Modeling, User Interface). Since the analysis software was bound into a single UNIX process, recompiling and relinking were required to tailor the software to the particular application. All users used the same code and had the same set of capabilities whether those capabilities were too few, just enough, or too many for the application.

Initially, Phase II was to follow the same architecture. However, it became apparent throughout Phase I and particularly at the Phase I demonstration that ICERVS had a broader application than originally envisioned. It was also apparent that there were several views of what ICERVS could be in addition to its role as input to robotic controllers. Some viewed ICERVS as a database; others saw it as a high level interface to sensors; while others focused on the integration of the database with the sensors. Almost everyone saw ICERVS as part of a solution and one element of an overall remediation system.

4.2.1. Guiding Principles

With the above considerations in mind, a refinement of the ICERVS architecture was initiated. In this process, several guiding principles were identified:

Consider the needs of both end users and system integrators of remediation systems: To date the focus has been on end users. Since system integrators of remediation systems have a major influence in the adoption of hardware and software, it makes sense to incorporate consideration of their needs in order to broaden the appeal for ICERVS.

Develop ICERVS to be consistent with DOE architecture for remediation systems: At present, the best information available is from the GISC and MiniLab systems being developed by Sandia. These systems use UNIX, Motif, and VxWorks to implement distributed processing over a network of computers. Software packages such as Sandia's GENISAS and ISOE provide a rich set of development tools in this type of environment. The preferred computers are Sun workstations, SGI workstations, and other VME-based computers.

Develop ICERVS to provide complete and general-purpose capabilities: The architecture and design for Phase II should be based upon filling gaps in potential users current systems. Identification of potential users and determination of their needs should become and remain a high priority.

Develop ICERVS as a set of independent system components: It is unlikely that DOE will use ICERVS by itself, but rather will incorporate it into an overall remediation system. Designing ICERVS as a set of independent but cooperative components makes it more flexible. Embracing a distributed architecture and the client-server model will make ICERVS integration into remediation systems more attractive to system integrators. The interface to each of these components should be flexible and extensible to permit growth as remediation system needs are better defined.

Implement ICERVS components as UNIX client-server programs: Each of the ICERVS components should be an independent CSCI implemented as a discrete UNIX program. These programs will interact with other system programs on a client-server basis, with the ICERVS programs functioning as servers. User written client software would provide the high-level operator interface and application sequencing logic. ICERVS could run locally or transparently on a remote computer. The client-server approach will go a long way toward (and may be mandatory for) ultimate acceptance of ICERVS in the DOE remediation community.

4.2.2. ICERVS Phase II Object Oriented Analysis (OOA) Model

This section will develop an initial OOA model for the ICERVS Phase II system. The model will be incomplete, but it will be a good starting point for defining the ICERVS Phase II system architecture. Later efforts will refine the OOA model. The paragraphs that follow define the problem domain, discuss what OOA is and is not, and then present the ICERVS OOA model.

4.2.2.1. What Is The Problem Domain?

A short (25 words or less) definition of the problem domain is:

ICERVS is a computer-based system that provides remote viewing, data acquisition, stored data visualization, data analysis, and model synthesis to support robotic remediation of hazardous environments.

4.2.2.2. What Is OOA?

Object-oriented analysis (OOA) is concerned with capturing the knowledge about the problem domain. It wishes to identify the classes (people, places, things) and the interactions (associations, messages, connections) between classes that pertain to the problem domain. It does not care about the human interface (windows, screens, etc.), nor the target computer environment (UNIX, DOS, etc.), nor a functional decomposition of the system, nor a data flow analysis of the system. These considerations come into play during design (human interface, computer environment) or are considered in a much more limited class-related scope during scenario evaluations (functional decomposition, data flow analysis). The classes are named using the ordinary terminology of the problem domain. This is an important contributor to the understandability of the OOA model. Additionally, the initial OOA model does not attempt to capture every detail; only the most important aspects are captured. The details are sought out and captured later as the problem domain becomes more clearly understood.

4.2.2.3. The ICERVS OOA Model

Figure 4-4 presents the OOA model for the ICERVS Phase II system. It identifies the major system level classes, whose names should be self-explanatory to those familiar with the waste site remediation problem domain. The diagram follows the Coad-Yourdon method, but uses slightly different symbology. Simple rectangles represent classes (attributes and behaviors are ignored). Various types of lines are used to capture the relationships between classes. A legend is provided on the diagram.

Most of the connections between classes have qualifiers attached. For example, the *Is Part Of* connection between **Waste Site** and **Sensor Subsystem** has a 0,m on one end and a 1 on the other end, which means:

1. A **Waste Site** contains from zero to many **Sensor Subsystems**.
2. A **Sensor Subsystem** is part of exactly one **Waste Site**.

Messages between classes are shown with the double line **Message Connection** symbol. A few of the message interactions could be:

1. A **Waste Site** can send a message to a **Sensor Subsystem** telling it to acquire data.
2. A **Waste Site** can send a message to a **Dataset** requesting data.
3. A **Sensor Subsystem** can send property data messages to a **Dataset** for storage.
4. A **CCV Camera** can send position messages to a **Dataset** for synchronizing its display.
5. A **Dataset** can send its **Geometric Object** information to the **World Model**.
6. **Architectural Plans** can send messages to insert themselves into a **Dataset**

Figure 4-4 ICERVS System Level OOA Diagram

The dotted line connection indicates specialization/generalization. For example, **Buried Pit**, **Underground Storage Tank**, and **Production Facility** are specializations of **Waste Site**. Each is a **Waste Site** and thus has all of that class's attributes and behavior. Each one is different from the others and thus has its own specific attributes and behaviors. **Waste Site** is a generalization of the other three classes. A similar specialization/generalization applies to **Sensor Subsystem** and its sub-classes.

4.2.3. Phase II Architecture Block Diagram

The software architecture for ICERVS Phase II is shown in Figure 4-5. The Phase II software will be composed of three primary Computer Software Configuration Items (ICERVS Phase II Demonstration CSCI, Volumetric Data CSCI, and Sensor Interface CSCI) and three sensor subsystem CSCIs (Structured Light Sensor CSCI, Laser Range Finder CSCI, and MiniLab CSCI). The Volumetric Data CSCI and the three sensor subsystem CSCIs act as servers. The ICERVS Phase II Demonstration CSCI acts as a client. The Sensor Interface CSCI acts as both a server and a client. Each CSCI is composed of one or more Computer Software Components. The following sections provide a brief description of each CSCI.

4.2.4. ICERVS Phase II Demonstration CSCI

The ICERVS Phase II Demonstration CSCI is the topmost application level process and contains a high-level operator interface to demonstrate the functions and features of the ICERVS Phase II system. This CSCI is representative of a user application and may be used as a template for creating a user application. A user application will generally be composed of a top level user interface and other software logic that implements the user's particular application. Each application must have an associated Application Data Structure organized and formatted in accordance with ICERVS requirements.

4.2.4.1. Top Level User Interface CSC

The Top level User Interface CSC provides the user with access to the Phase II main level functions. It will incorporate some evolutionary upgrades (such as menu revisions) to the existing Phase I graphical user interface. It will also add a new capability for generating and managing sensor windows. It is anticipated that the bulk of user interface revisions will be in response to the needs of actual site operators. Figure 4-6 illustrates the ICERVS Phase II top level user interface screen and menus. Five major menu categories are implemented:

System: Provides the ability to log onto the ICERVS system and to exit the ICERVS program.

Site: Provides menu functions that support the addition and deletion of sites (workspaces), and the editing and printing of site parameter and site log files. In Phase I, these functions were part of the **Model** menu, which has been renamed to be more domain specific.

Volumetric Data: Provides the ability to link the application to the ICERVS Volumetric Data Server. If an interactive connection is made, the Volumetric Data Server will display a control window that permits interaction with the application's volumetric database. The functions provided include opening, closing, and saving datasets; creating and manipulating views; inputting data; performing analysis functions; and defining and manipulating cut planes.

Figure 4-5 ICERVS Phase II Analysis Software Architecture

Figure 4-6a -- ICERS Phase II Top Level Screen and Menu

Figure 4-6b -- Volumetric Data Window Screen

Connections: Provides menu functions for establishing, breaking, and inspecting connections between sensors and datasets. Also included is the capability to synchronize a view of a volumetric dataset to a CCV camera subsystem so that the same scene is displayed by both.

Sensor: Provides menu functions to support the opening, closing, addition, deletion, editing, printing, and inspection of sensors on a site by site basis.

Help: Provides menu functions for supplying help information. Phase II will not implement most of the menu functions.

4.2.4.2. Application Data Structure

The data manipulated by the ICERVS Volumetric Data Server and ICERVS Sensor Interface Server belong to the end user and are stored in some user data storage area. When an end user activates the ICERVS package, the location of the user's data must be specified. Furthermore, in order for the ICERVS servers to access and manipulate the data, the organization and contents of the Application Data Structure must be standardized. The general organization for the Application Data Structure is as follows:

/usr/SOMEWHERE/...	An arbitrary point in users disk hierarchy
/ICERVS	Root of ICERVS Data Structure Contains template files and default files
/etc	Contains general control files
/SITE1	Contains parameters files for first site, raw data files, and volumetric/geometric data files
/SITE2	Contains parameters files for second site, raw data files, and volumetric/geometric data files
.	
.	

4.2.5. Volumetric Data CSCI

The Volumetric Data CSCI encapsulates the storage, retrieval, manipulation, and visualization of spatial, property, and geometric object data. It is implemented as a UNIX server. Six of the CSCs (excepting the Visualization/Interaction CSC) form the core of the ICERVS Volumetric Data Server. The Visualization/Interaction CSC will act as client of the ICERVS Volumetric Data Server. Figure 4-7 illustrates the Volumetric Data CSCI and its CSCs. The double-line connections represent the path followed by commands to the server. The single-line connections represent inter-CSC function calls within the CSCI. The following sections briefly describe each constituent CSC.

Figure 4-7 Volumetric Data CSCI Block Diagram

4.2.5.1. Volumetric Data Client Interface CSC

This CSC incorporates the Sandia ISOE software and the ICERVS client-server communications software. The CSC is responsible for accepting commands from client applications, parsing the command messages, validating the command, and dispatching the command to the server command execution functions. These execution functions will typically call upon routines in the other CSCs that comprise the Volumetric Data CSCI. Included in this CSC are functions to send reply messages to the client after the command has been processed.

4.2.5.2. Spatial Data Engine CSC

The Spatial Data Engine (formerly Octree Engine) CSC provides the primary means for storing, retrieving, and manipulating volumetric data. It will incorporate a major upgrade in the capabilities for integrating and analyzing measurement data. The key to this will be the inclusion of Octree Corporation's TrueSolid software. In addition to providing a controlled interface to the TrueSolid software, this CSC will provide enhanced features such as general cut plane tools for controlling display and sculpting; and spatial comparisons between measurement data sets. The Spatial Data Engine CSC is also the gateway to the Property Data CSC.

4.2.5.3. Geometric Object CSC

The Geometric Object (formerly Object Modeling) CSC provides for the creation, modification, and storage of geometric objects. It will include upgrades to the tools for creating geometric objects. This will include more general prismatic shapes and geometric primitives such as rectangular parallelepipeds, cylinders, and spheres. The internal representation of the object will be upgraded to more closely resemble the IGRIP interchange format.

4.2.5.4. Spatial Data / Object Interactions CSC

The Spatial Data / Object Interactions (formerly Octree/Object Interactions) CSC provides a link between the Spatial Data Engine CSC and the Object Modeling CSC. It will incorporate capabilities to enable mathematical operations between octree data and geometric object data. The CSC will allow an operator, for example, to compute the volume of octree material located inside a selected geometric object and automatically scan octree space to identify any geometric objects that are void (contain no octree material).

4.2.5.5. World Model Interface CSC

This CSC will convert ICERVS geometric objects created by the Geometric Object CSC into a suitable format to provide geometric world models to robotic controllers in a specified form such as IGES, STEP, or IGRIP. (Phase II will support the IGRIP format.) Future capabilities will support input of geometric object information from CAD drawings and architectural plans and possible integration with a robotic controller world model.

4.2.5.6. Property Database CSC

This CSC has been separated from the Spatial Data Engine CSC to promote system flexibility and modularity. During Phase II, it will remain somewhat coupled to the Spatial Data Engine CSC. To accommodate the large amounts and multitudinous varieties of property data in an efficient manner, it is anticipated that a commercial DBMS product will be integrated into the Property Database CSC in the Phase III ICERVS. This CSC and the DBMS will provide for the physical storage/retrieval of property data, while the Spatial Data Engine will be used to provide the spatial aspects of the property data. The Property Database CSC will provide some simple data manipulation functions.

4.2.5.7. Visualization and Interaction CSC

This CSC has been separated from the Spatial Data Engine CSC to promote system flexibility and modularity. During Phase II, it will remain somewhat coupled to the Spatial Data Engine CSC. Enhancements to the visualization of volumetric data will be provided, such as, arbitrary viewpoint, orthogonal viewpoints, and pseudo coloring based on property data. Enhanced geometric object manipulation capabilities will be added to the software. The display of geometric objects will be user selectable as wire frame or shaded surface. This CSC is expected to incorporate portions of UNIRAS ToolMaster agX and Rogue Wave Canvas. In addition, it will interact with the Spatial Data Engine to access TrueSolid functions.

Visualization and interaction are accomplished via view windows (Figure 4-8b), which may be created by the operator from the Volumetric Data Window VIEW menu (Figure 4-8a). View windows display volumetric and object information that represents the 3D surface and objects of a waste site. Both orthogonal and arbitrary views are supported. The types of view window(s) that initially come up are defined in the view defaults file. View windows contain two scroll bars for translating (panning) the image and another scroll bar for scaling the image. Arbitrary views also contain a third scroll bar for translating in the third dimension as well as controls for rotating the image about all three coordinate axes. The view window supports the following menu functions:

Display	Object	Options
<u>Display:</u>	Provides several display related control panels for setting colors, view type, plot type, and other options; numerically specifying transformations (translation, rotation, and scaling); and obtaining printed maps.	
<u>Object:</u>	Displays a control panel that permits creation, deletion, printing, export, manipulation and editing of geometric objects.	
<u>Options</u>	Provides several ancillary functions that are not required parts of the Phase II system, but which facilitate the development and testing of the system.

Figure 4-8a — Volumetric View Window

Figure 4-8b - View Data Window

4.2.6. Sensor Interface CSCI

The Sensor Interface CSCI manages the interface to the ICERVS sensor subsystems and provides functions common to all sensor subsystems, such as Add Sensor, Delete Sensor, Connect Sensor, Disconnect Sensor, etc. In the future, this CSCI will provide an automated data acquisition capability and tools for converting the sensor data into a form compatible with the volumetric database. (The Phase II system will implement these capabilities in the Structured Light Sensor CSCI and will require the Laser Range Finder and MiniLab Simulators to provide spatial data.). The Sensor Interface CSCI is implemented as a UNIX server. In addition, it is a client of each attached sensor subsystem server. Figure 4-9 illustrates the Sensor Interface CSCI and its CSCs. The double-line connections represent the path followed by commands to the server. The single-line connections represent inter-module function calls within the CSCI. The following sections briefly describe each constituent CSC.

4.2.6.1. Sensor Client Interface CSC

This CSC incorporates the Sandia ISOE software and the ICERVS client-server communications software. The CSC is responsible for accepting commands from client applications, parsing the command messages, validating the command, and dispatching the command to the main server command execution functions. These execution functions will typically call upon routines in the other CSCs that comprise the Sensor Interface CSCI. Included in this CSC are functions to send reply messages to the client after the command has been processed. This CSC is very similar to the Volumetric Data Client Interface CSC (paragraph 4.2.5.1) and will be designed to maximize commonality with that CSC.

4.2.6.2. Sensor Subsystem Server Interface CSC

This CSC incorporates the Sandia ISOE software and the ICERVS client-server communications software. The CSC is responsible for making connections to external sensor subsystems, formatting client commands, sending client commands, and receiving replies and sensor data from the sensor server.

4.2.6.3. Sensor Data Interface CSC

This CSC provides a data path between the sensor and one or more data destinations. Possible destinations are the client application, the Volumetric Data Server, a data processing server, or an archive file. The data stream from the sensor may be buffered to accumulate data for batch transmission to the destination. During Phase II, all sensor data will be routed to the client application, which will then route the data as required. In Phase III, it is envisioned that the Sensor Data Interface CSC will directly route the sensor data to one of the possible destinations and perhaps multiple destinations simultaneously.

4.2.6.4. Sensors Database CSC

The Sensor Database CSC provides functions for adding, deleting, assigning, and modifying sensors and cameras at both the site level and the system level. Phase II will implement only the site level support functions; system level functions will be performed off-line. Phase III will provide full support for adding, deleting, and modifying sensors at the system level. The data describing sensors is stored in the Application Data Structure and belongs to the client application. This CSC merely maintains the data on the client's behalf.

Figure 4-9 Sensor Interface CSCI Block Diagram

4.2.6.5. Automated Data Acquisition CSC

This CSC is planned for the future and will provide an automated data acquisition capability. It will incorporate and generalize the capabilities from the Operational Sequencing CSC in the Structured Light Sensor CSCI.

4.2.6.6. Sensor Data Toolkit CSC

This CSC is planned for the future and will provide tools for converting the sensor data into a form compatible with the volumetric database. These tools will take the form of data filters, interpolators, transformations, and other pre/post processors. Three possible scenarios are envisioned:

1. sensor ==> Toolkit function ==> database
2. raw database ==> Toolkit function ==> processed database
3. two property datasets ==> Toolkit function ==> derived property dataset

In the first scenario, a sensor is connected to a database with an intervening data processing function. In this way, sensor data flows from the sensor, is transformed into spatial data, and is inserted into the data base. In the second scenario, raw data has previous been stored in a database. The database is later connected to a processed database through a toolkit data post-processing function. The raw data is transformed to extract a particular type of information. It may be meaningful to transform the raw data via several different processing algorithms to extract different information from that single raw data set. The third scenario illustrates a type of data synthesis whereby two sets of previously stored property data are extracted from a database, combined, and entered into the same or a new database as a derived property.

4.2.7 Sensor Subsystem CSCIs

A Sensor Subsystem CSCI manages a sensor subsystem and interfaces to the ICERVS in accordance with the Common Interface for Sensors specification. Each sensor subsystem CSCI is implemented as a UNIX server. Figure 4-10 illustrates a Generalized Sensor Subsystem CSCI and its CSCs. The double-line connections represent the path followed by commands to the server. The single-line connections represent inter-module function calls within the CSCI. The Generalized Sensor Subsystem CSCI is conceptual and is not a required architecture for a sensor subsystem. The number of CSCs implemented and their functionality will vary for each sensor subsystem. The only requirement imposed by ICERVS is that the interface to the sensor subsystem comply with the Common Interface for Sensors specification. The following paragraphs briefly describe the constituent CSCs for the Generalized Sensor Subsystem CSCI.

Sensor Subsystem Client Interface CSC: This CSC is responsible for accepting commands from client applications, parsing the command messages, validating the command, dispatching the command for execution, and sending the reply message after execution completes. This CSC must comply with the Common Interface for Sensors.

Figure 4-10 Generalized Sensor Subsystem CSCI Block Diagram

Control Panel CSC: This CSC is the user interface for the sensor subsystem. The control panel must be XWindows compatible so that it may be displayed on the ICERVS computing platform screen. The ICERVS operator will control the sensor subsystem by means of this control panel.

Sensor Hardware Interface CSC: This CSC provides the interface with the sensor hardware.

Data Processing CSC: This CSC performs any necessary data reduction and processing to convert raw data into sensor data.

Sensor Simulator CSC: This CSC will simulate the operation of the sensor subsystem. It may be controlled from the Control Panel CSC, but typically will implement its own control panel and will incorporate the Sensor Subsystem Client Interface CSC also.

Operational Sequencing CSC: This CSC will provide capabilities that enable automatic operations to be carried out with the sensor. The CSC will allow an operator to define a set of control parameters for a workspace and will subsequently command the sensor subsystem to measure data without further operator involvement. It is expected that, during ICERVS Phase III, common elements from this CSC for several surface mapping sensor subsystems will be generalized and incorporated into the Automated Data Acquisition CSC for the Sensor Interface CSCI.

Dataset Registration CSC: This CSC implements the algorithms that allow two or more sets of data to be aligned, registered, and merged into a composite dataset. It is expected that, during ICERVS Phase III, common elements from this CSC for several types of sensors will be generalized and incorporated into the Sensor Data Toolkit CSC for the Sensor Interface CSCI.

4.2.7.1. Structured Light Sensor CSCI

As part of CRADAs with Sandia and Oak Ridge, MTI is developing a 4" diameter, single station version of a structured light sensor, including the controller for the sensor. This CSCI integrates the controller, the controller software and the ORNL developed mapping sensor control software. This CSCI will reside on the Sun Classic and will implement only 4 of the 7 generalized CSCs. Much of this CSCI will have been developed under the CRADA II effort. The following CSCs will be implemented:

Sensor Subsystem Client Interface CSC: This CSC performs as described in paragraph 4.2.7. It is very similar to the Volumetric Data Client Interface CSC (paragraph 4.2.5.1) and will be designed to maximize commonality with that CSC.

Sensor Hardware Interface CSC: This CSC interfaces with the structured light controller computing platform (a 486/DOS personal computer) and the structured light controller software. The controller software provides for the calibration, registration, and control of the instrument stations; processing of image data; calculation of 3D measurement results; and interface with the

workstation based sequencing code. A large part of this CSC will be developed under the CRADA II effort.

Operational Sequencing CSC: This CSC will incorporate capabilities that enable automatic operations to be carried out with the structured light sensor. This CSC will allow an operator to define a set of regions or paths to be scanned in a workspace and will subsequently command the sensor subsystem through the set of regions or paths so that measurement data is acquired without further operator involvement. A large part of these capabilities is contained in the ORNL structured light mapping sensor code.

Control Panel CSC: This CSC is the user interface portion for the structured light sensor. The Sun Classic version of this user interface will have been developed by ORNL under CRADA II. The ICERVS effort will upgrade the ORNL code to display on with the SGI-based computing platform and to be compliant with the ICERVS Common Interface for Sensors specification. Once opened, the control panel provided by this CSC will be used to interact with the sensor.

4.2.7.2. Laser Range Finder CSCI

Only the Sensor Subsystem Simulator CSC will be implemented for this CSCI. The CSCI will simulate the PNL developed laser range finder sensor on the Sun Classic. MTI will design a remotely interfaced CSCI in accordance with the ICERVS Common Interface For Sensors standard. As of September 1993, it is not known if PNL will provide source code and/or test data for incorporation into the Phase II ICERVS. If available, actual measurement data from PNL will be incorporated into the simulation; otherwise, a mathematical model will be used to generate appropriate data. If actual measurement data are used, this CSCI will implement the necessary data transformation and registration algorithms to make the data compatible with the volumetric database. At a future point, these algorithms will be considered for incorporation into the Sensor Data Toolkit CSC of the Sensor Interface CSCI.

4.2.7.3. Sensor MiniLab CSCI

Only the Sensor Subsystem Simulator CSC will be implemented for this CSCI. The CSCI will simulate MiniLab on the Sun Classic. MTI will design a remotely interfaced CSCI in accordance with the ICERVS Common Interface For Sensors standard. It is expected that test data from INEL will be available in November 1993, and will be incorporated into the simulation; otherwise, a mathematical model will be used to generate material property data, such as temperature or electrical conductivity. If actual measurement data are used, this CSCI will implement the necessary data transformation and registration algorithms to make the data compatible with the volumetric database. At a future point, these algorithms will be considered for incorporation into the Sensor Data Toolkit CSC of the Sensor Interface CSCI.

4.3. Allocation of System Requirements

Table 4-1 assigns the system requirements to the various CSCIs and the structured light HWCI.

Table 4-1 Allocation Of System Requirements To ICERVS Phase II CSCIs

CSCI	Full Implementation	Partial Implementation	Phase II Design Influence
Phase II Demo Application CSCI	R3.12, R6.01, R6.04, R6.05, R6.07, R6.08, R6.09, R8.01		R8.02, R8.03
Volumetric Data CSCI	R1.01, R1.02, R1.03, R1.04, R1.05, R1.06, R1.07, R1.08, R1.11, R2.01, R2.02, R2.05, R2.06, R2.07, R2.08, R3.01, R3.02, R3.03, R3.04, R3.05, R3.06, R3.07, R3.08, R3.09, R3.10, R3.11, R3.13, R5.01, R5.02, R5.03, R5.04, R5.05, R5.06, R5.07, R5.08, R5.11, R6.02, R6.03, R7.06 R7.07	R1.02, R2.03, R1.10, R3.14	R1.09, R2.04, R5.09, R6.06
Sensor Interface CSCI	R7.01, R7.02, R7.03, R7.04, R7.05, R9.07	R5.10	
Structured Light Sensor CSCI	R9.01, R9.04, R9.05, R9.06	R9.02, R9.03, R5.10, R7.08	R4.02
Laser Range Finder Simulator CSCI		R5.10, R7.08	
MiniLab Simulator CSCI		R5.10, R7.08	
Structured Light Sensor HWCI		R4.01, R9.10, R9.11	R4.03, R4.04, R10.01, R10.02, R10.03, R10.04

5. COMMON INTERFACE FOR SENSORS

To acquire data the ICERVS system must be attached to a sensor. This could be achieved in many different ways. Since a loose coupling is desired, a client-server relationship between ICERVS and a sensor provides an elegant solution. A client-server relationship requires an interface specification which is provided by the Common Interface for Sensors (CIS). The CIS embodies a communications protocol and an Applications Program Interface (API). This interface is used by both clients (ICERVS application) and servers (sensor applications) in the ICERVS system for sensor control and data acquisition.

5.1 General Requirements

In general an ICERVS sensor is an intelligent sensor system. That is, it is a sensor system that performs complex functions in response to high level commands and provides data that has been reduced to some degree. Examples of intelligent sensor systems would be a structured light system or a laser ranging system. The CIS applies only to these intelligent sensor systems. Any simple sensor data (e.g. temperature) is envisioned to be provided by MiniLab. MiniLab is a data acquisition and robot control system for use in waste site remediation.

The communications protocol portion will use the Intelligent System Operating Environment (ISOE) developed by Sandia National Laboratories for their Generic Intelligent System Control (GISC) kit. This is essentially a package to ease the use of socket based communications in UNIX systems. Sockets are widely used as the communications method for clients and servers under the UNIX operating system. The API portion of the CIS contains a small set of standard commands that a server must implement. The standard command set may be extended by adding sensor unique commands required by a server.

Most intelligent sensors will have a graphical user interface (GUI) that is used on the sensor's local computer (e.g. sensor control panel). The ICERVS system will allow this GUI to be used on a remote computer. Once connected to a server, ICERVS may request the sensor to execute it's GUI. At this point the GUI will be displayed on the ICERVS computer display (a remote display with respect to the sensor system). In order to operate properly, the sensor's GUI must be X windows compatible, but Motif is also acceptable.

5.2 Communications Protocol (ISOE / GENISAS)

The communications protocol to be used for CIS is UNIX socket based so that a sensor may be controlled by a remote system (i.e. the ICERVS platform) over Ethernet, or locally on the same platform. Use of GENISAS / ISOE will allow attention to be paid to development of the interfaces instead of the low level details of communications.

A typical connection process only requires the client to know the host name of the system that the server is on, and the service name as listed in a well known database of service names (e.g. the /etc/services file in UNIX). Once this information is known the client uses its connection command(s) to initiate a connection with a server.

5.3 Applications Program Interface

The API consists of a set of standard and optional commands that allow a client to communicate with a server. The standard commands must be implemented by any client/server using the CIS. As the name implies, the optional commands are not required.

5.3.1 Standard Commands

The following list describes the standard commands that are required by the CIS.

INIT, selftest

selftest = 1 or 0

The INIT command will cause the server to put the sensor system in its initial state. This does not imply that the client needs knowledge of the various states of the sensor system. The intended use is to start the sensor from a known condition or recover from an error condition. The self test option is a flag that will cause the sensor to run it's onboard self test. If the sensor does not have a self test, any returned status will indicate that the test passed. Self test status can be acquired by use of the STATUS command.

STATUS

The STATUS command will return the current status of the server including the results of the most recently executed self test. The data is returned as an ASCII string.

START

The START command will cause the sensor to begin it's sensing function. This means that the client is expecting to receive data and will continue to take input from the server until it issues a STOP command.

STOP

The STOP command will cause the server to stop sending data and any other functions it is currently performing.

Admittedly these commands are very simplistic, but this is desired to keep the standard interface easy to use. More sophisticated commands may be added by the developer of the server as optional commands.

5.3.2 Optional Commands

Optional commands are implemented as needed by each sensor system to accomplish their mission. These commands could also include server specific commands as deemed necessary by the designer.

START_GUI.display

display = X windows compatible display (e.g. hostname:0)

The START_GUI command will cause the sensor system to start it's X windows based GUI on the specified display. This is strongly encouraged since all or most of the server specific operations may be performed by use of this GUI.

5.4 Sensor configuration requirements

Sensor configuration data will consist of all data that is necessary for the client to access and use the sensor server. This data will be provided by the developer of the server as a sensor configuration file. The configuration file is an ASCII file with the following format:

[SectionName]
ConfigItem=ConfigString

This file will be read by a utility that configures the CIS for the new server (e.g. -- this utility could be accessed from the ICERVS user interface). A sample file is shown below that contains the necessary configuration items. The file contains two sections, required configuration items, and optional configuration items.

[Required]	
ServerName=mti_slms	Service name to be put in /etc/services file
TCPPortNumber=1111	TCP port number to be put in /etc/services file
DataPreamble=	Format of data at the start of a data block, if any
DataLineFormat="%lf %lf %lf"	Format of a line of data from the server
DataPostamble=	Format of data at the end of a data block, if any
GUI=Yes	Boolean to indicate if the server supports a GUI
[Optional]	
NumCameras=2	Example of optional configuration item

The optional configuration items section is not required to be in the file if optional items do not exist.

THIS PAGE INTENTIONALLY LEFT BLANK

6. SOFTWARE DESIGN CONSIDERATIONS

6.1. Object Oriented Methodology

The ICERVS software will be designed and implemented using an object-oriented approach. Key to the object-oriented approach is the concept of the class. The class provides a mechanism for combining the data and the manipulation procedures related to a high-level entity into a single construct. Classes facilitate abstraction (ignoring details of processes and how data is represented), promote encapsulation (hiding of the internal workings of entities) and support inheritance (defining a new entity as a specialization of another entity). The first step in object-oriented analysis and design is to identify the classes. Later steps involve assignment of attributes and behavior, identification of relationships between classes and arrangement of the classes into hierarchies.

During object-oriented analysis (OOA), the system analyst is concerned with the Problem Domain Component (PDC) classes. The PDC classes correspond to the real world objects (people, places, or things) that one must deal with and manipulate in the normal course of problem solving within a particular application domain. The problem domain, its objects, and the classes that represent the objects are relatively stable over time. As such, problem domain based classes offer a reasonable assurance of being reusable. The analysis and design of the ICERVS CSCIs will begin with the identification of the major PDC classes and their the associations and inter-connections. A form of the Coad-Yourdon documentation method will be used to document the analysis for each ICERVS CSCI.

During object-oriented design (OOD), the system designer takes the PDC analysis and adds more classes to complete the design. One group of classes, the Human Interface Component (HIC) classes, tends to be very volatile. HIC classes are concerned with the human interaction aspect of the problem domain and, as such, it is easy to see why they are very volatile over fairly short periods of time. (e.g., Consider user interfaces that have run the gamete from question/answer oriented, to text menu-driven, to GUI, and are soon to be supplanted by object-oriented methodologies.) A very important aspect of OOD is the rigid control of the interactions between the PDC and HIC classes. A PDC class should know little if anything about its corresponding HIC classes. In this way, if the HIC environment changes, there is minimal impact on the PDC classes. The HIC classes, however, are intimately knowledgeable of their corresponding PDC classes. Since the PDC classes are very stable, there will be minimal impact due to changes. This concept is best expressed by the SmallTalk Model-View-Controller architecture. The design of the ICERVS CSCIs and their constituent CSCs will pay particular attention to the Model and View component aspects. As can be seen from the ICERVS architecture presented in section 4.0, the separation between the Model and View components has already begun.

ICERVS will continue the object-oriented approach into the programming of the software. The C++ language will be used to implement the ICERVS software.

The object-oriented methodology does not follow a straight line path from OOA through OOD to OOP. Rather, it is more of an iterative process. One starts by creating an initial OOA model and follows that with an initial OOD. Some OOP may be performed next to test key concepts in the model and to help

clarify design alternatives. The process then most often loops back to OOA and OOD to refine the model. This OOA-ODD-OOP iteration continues until the model is complete. The final OOP can then be completed.

The process outlined above is gradually developing deeper and clearer understanding of the problem domain at each iteration step. Frequent tangible results can be demonstrated every few weeks during the process.

The advantages in using OOA, OOD and OOP for the ICERVS software are uniformity, understandability, flexibility, stability, and reusability.

6.2. Client-Server Communications

ICERVS Phase II will embrace a client-server architecture. At this time, the SNL Intelligent System Operating Environment (ISOE) software is the probable candidate to provide the client-server communications. The SNL GENISAS package is still being investigated.

The long range architecture for ICERVS envisions the need for multi-client servers and combined server/client processes. (The Phase II design will not implement the multi-client servers, however, the design will not disallow the future addition of this feature.) These needs do not appear to be immediately supported by ISOE under UNIX. Further investigations are needed. Secondly, a C++ wrapper over ISOE will be needed. GENISAS provides the C++ wrapper support, but may present too much processing overhead to be practical for ICERVS.

An overriding consideration is future GISC compatibility. To be accepted into the DOE community, GISC compatibility may be a requirement.

6.3. Purchased Software

Several software tools will be purchased for use during the Phase II software design and implementation processes, including:

1. UNIRAS ToolMaster
2. Rogue Wave Canvas
3. TrueSolid.

The use and incorporation of these software tools into ICERVS will reduce the effort and implementation cost of the software. A brief description of each software product follows.

A large part of the ICERVS software is dedicated to the implementation of the various menus, buttons, controls, display windows, and other GUI components. The UIM/X part of UNIRAS ToolMaster will allow the interactive specification of Motif windows, widgets, and gadgets. Once specified, the UIM/X package will automatically generate the Motif-specific aspects of the GUI code in the C language. Since these functions are C language functions, C++ wrappers will be required for isolation and compatibility

with the remaining ICERVS software. This tool should significantly reduce the effort involved in implementing the user interface portions of the ICERVS software. The Rogue Wave Views++ user interface code implemented in Phase I is Motif compatible and thus can coexist with the UIM/X generated code.

UNIRAS ToolMaster agX provides a comprehensive 2D and 3D graphics library. This library is built on X-Windows and Motif and is thus compatible with other aspects of the ICERVS user interface. High level functions for 3D images including rotation, translation, hidden line removal, cut planes, and etc. are provided. The functions provided by agX are C language functions and will thus require C++ wrappers for isolation and compatibility with the remaining ICERVS software. ToolMaster agX support for interactive graphics is unknown at this time.

Rogue Wave Canvas is a 2D and 3D graphics support package. It is implemented as a library of C++ classes. Interactive creation and manipulation of 2D and 3D geometric objects is provided. Compatibility with ToolMaster is expected since Canvas is also based on Motif. One drawback to Canvas is that it is not due for release until late in the Phase II effort. However, alpha-release copies of Canvas are being obtained in order to evaluate its potential application with ICERVS.

TrueSolid, from Octree Corporation, is an octree-based data storage and visualization package. It supports many of the data visualization and analysis features required by ICERVS Phase II. Incorporation of this software product into ICERVS is presently being evaluated. The lack of advanced interactive graphics and geometric object modeling capabilities are two potential short comings of the TrueSolid software.

The final design of the ICERVS Phase II software will probably incorporate each of the above to some extent. Each package has advantages and the packages complement each other very well.

THIS PAGE INTENTIONALLY LEFT BLANK

7. TESTING

7.1 Test approach/philosophy

Because the Phase II ICERVS software is not life critical, testing will be informal, yet thorough. At the lower levels of design, class and CSC test procedures, test software and test cases/data will be designed and implemented by the responsible class/CSC designer(s) with guidance from the ICERVS lead programmer. The class and CSC tests shall be documented to the level needed for an ICERVS programmer to run the tests and evaluate the results without undue experimentation or uncertainty. Much of the user interface software can be tested within the UNIRAS ToolMaster UIM/X package, thus reducing the need for the development of special testing routines. The preferred test documentation is descriptive comments in the applicable source files. In general, each class and each CSC shall be tested prior to higher-level integration, although exceptions may be approved by the ICERVS lead programmer.

At the CSCI and system level, test procedures and test cases will be designed and implemented by the ICERVS system engineer. An individual procedure will be documented for each CSCI. The procedure will be designed so as to verify the ICERVS requirements assigned to each CSCI.

7.2 CSCI Testing

7.2.1 Demonstration CSCI Testing

The demonstration CSCI will be implemented so that it can be operated with or without servers available. Initial testing will be standalone, i.e., without servers available, and will verify menu traversal and control panel operation to the extent possible. Testing will also include attempts to connect to (unavailable) servers, with the appropriate error messages displayed to the user.

7.2.2 Volumetric Data CSCI Testing

If practical, the demonstration CSCI will be used as a front end to run and test the volumetric data CSCI. The volumetric data CSCI (server) will be started and run through its capabilities. Test data will include the Fernald silo data with vertical features (e.g., pipe and wall sections) added by MTI. Data from the CRADA system, laser range finder, and/or buried waste MiniLab will also be used if available. The test procedures will emphasize the new features added in Phase II, particularly: arbitrary viewpoint, arbitrary cut planes, matching scene display ("snap" feature), advanced polyhedral shapes for geometric objects, geometric object primitives, shaded-surface display of geometric objects, computation and display of volumetric data inside geometric objects, and IGRIP file generation. A separate procedure will be used to test material property capabilities that derive from the INEL add-on task, including: data interpolation, temporal pattern computation and display, and multiple-sensor pattern computation and display.

7.2.3 Sensor CSCI Testing

This section covers testing of the Structured Light CSCI, the Laser Range Finder CSCI, and the MiniLab CSCI. Each CSCI is a server that is to conform to the ICERVS Common Interface for Sensors. To verify this conformance, a special test program will be written. This test program will establish an ISOE connection to a user-provided server name, and provide the user with control panel buttons to send

standard sensor commands. The test program will also provide a capability for sending user-provided command messages to the server CSCI to test sensor-specific features. (If practical, these sensor-specific commands can be provided either by keyboard entry or from a text file, with commands read in one at a time.) The test program will also provide a text display of messages received from the server CSCI.

A prescribed sequence of standard commands will be defined for verifying conformance to the common interface. The test program will initially run on the Sun computer, which is the same platform as the sensor CSCIs. For final sensor CSCI testing, the test program will be run on the SGI platform and the sensor CSCIs will be run on the Sun.

After testing the common interface conformance, each sensor CSCI will be tested for its sensor-specific capabilities. The test program will be used to initiate the sensor user interface, and the sensor-specific capabilities will be tested through that interface.

7.3 System Testing

System testing will be accomplished by successfully completing the system demonstration procedures that will be used to present ICERVS to external personnel. The demonstration procedures presently include: Structured Light Sensor, Volumetric Data Base, Laser Range Finder, Material Properties, Integrated Operation, and IGRIP. Some variations and additions to the procedures may be made during testing to assure completeness and robustness.

Appendix A: ICERVS Phase II Mission Profile Revisions

As part of the ICERVS preliminary design, mission profiles were described for three remediation tasks. The original profiles are documented in Section 3.0 of the ICERVS Phase I System Design Report. During the initial stages of the Phase II design effort, it became apparent that some minor revisions in the mission profiles were necessary in order to clarify points or to reflect the greater insight into the needs of the DOE community. A summary of the revisions to the mission profiles is given below and an updated copy of the Mission Profiles, including the revisions, is contained in Appendix B.

General:

Changed the term *model* and *object model* to *geometric object*.

Underground Storage Tanks (Section 3.1):

Added new section (3.1.7) describing data archival activities.

Buried Waste Sites (Section 3.2):

Added statement to section 3.2.1 that some missions will be a surgical retrieval from a localized region instead of large-scale retrieval of the entire site.

Inserted new section (3.2.2) describing installation activities.

Added a new paragraph to section 3.2.3 describing the use of material property sensors to predict underlying objects beneath the dig face.

Production Facility D&D (Section 3.3):

In section 3.3.1 and 3.3.2 added the digitization of existing facility drawings and their on-line availability to operators.

In section 3.3.2 provided a more detailed description of object modeling activities.

THIS PAGE INTENTIONALLY BLANK

Appendix B: Revised Mission Profiles

This appendix provides an updated copy of the ICERVS Mission Profiles including the revisions made as part of the Phase II system design effort. In order to avoid confusion, the paragraph number is the same as the ICERVS Phase I System Design Report.

B.3.0 Mission Profiles

This Section describes several potential applications for the ICERVS in the DOE Environmental Restoration and Waste Management Program. Three missions of broad scope were chosen in order to assure generality and the selection of a suitable architecture.

These missions are described primarily from the point of view of surface mapping activities. As was noted above, the resulting set of requirements will be broader in scope than the current contract.

B.3.1 Underground Storage Tanks

Waste streams from production processes have been stored in underground tanks at various sites. The remediation of these tanks includes a characterization of their interiors and the removal of their contents. ICERVS is one of the tools that an operator may use to assist him/her in carrying out these tasks.

B.3.1.1 Overview

The operator is assumed to be at least an entry-level engineer who has received appropriate training on the system and for this application. The operator is located in a trailer or equipment shelter at some safe distance from the tank. S/he views the actual operation via a closed-circuit video camera that are attached to the robotic system(s), mapping sensor(s), or have been placed in the tank as standalone viewing equipment. A computer in the trailer or shelter provides the user interface for operation of the mapping sensor and other systems. The operator interface equipment includes the following.

- Video monitors for the closed-circuit video cameras
- Color graphics computer monitor, window-based
- Keyboard, spaceball and/or mouse
- Communications link (network, modem, etc.)
- Data storage/archival equipment (VCRs, video scan converters, tape drive, WORM, etc.)
- Optional printer and/or plotter

To permit concrete discussions in this application, it is assumed that the sensor subsystem contains a structured light mapping sensor. In fact, any line of sight sensor could be used.

B.3.1.2 Installation

Installation contains both physical and logical elements. The site must be prepared for the equipment to be used and the necessary permits secured. The selected equipment must have been verified and validated

before being brought to the site and installed in the tank. Power and communication links must be established and tested and the basic functioning of the system verified.

The logical elements to be installed in the computer include the known tank plans — particularly the location of all risers to be used for deploying the mapping sensors, the locations of any registration targets, core samples or related data, and the sculpting of known regions in the volumetric representation of the tank interior.

B.3.1.3 Registration

For accurate measurements, it is necessary to know the precise location of the mapping sensor inside the tank. Special targets may be needed to carry out this "registration". The operator will aim the sensor at each of the registration targets and the mapping sensor elements will perform any required measurement and computation. The result will be a message from the mapping sensor reporting its location in the tank coordinate system. Registration is complete when all the mapping sensor positions have been reported to the operator.

B.3.1.4 Mapping

The mapping process begins with a visual inspection to plan the mapping followed by a detailed inspection to gather the necessary data.

The operator will begin by visually inspecting the tank interior, using the closed-circuit video cameras and/or the mapping sensor video cameras. Pan and tilt positions will be commanded through a spaceball/mouse in the form of rate commands. The joint angles at the viewing station will be continuously updated to support the display for the operator. This display will include mapping sensor positions, current azimuth angle of the viewing camera and any architectural plan data available. Other views of the tank are available to the operator to display architectural features, the position of the mapping sensors and registration targets, and any recorded core samples. The operator can pull up descriptions of these features by clicking on the screen. S/he can record any observations and notations in a computer data base (electronic notebook).

The next step is to develop the mapping strategy. The vertical surfaces will probably be mapped before the horizontal surfaces to identify any vertical elements that could obscure segments of the horizontal waste surface to be mapped. Operational parameters will be selected, to include:

- Image parameters (spacing between images, number of points per image, nominal surface locations, such as Z height)
- Display parameters (pseudo-color map per view, engineering units, object format such as wire frame or shaded polygon)

The operator will plan the paths by drawing/displaying lines on the plan /side view and using cut planes to establish the view region.

After the operator has completed the path planning, s/he will issue a command to begin the actual mapping. The system will automatically update the pseudo-color plan/side view to indicate the areas that have been mapped. The raw data will be automatically backed up at regular intervals. The operator will inspect the resulting data, using both the display and the computer values. Of concern are any missing data, any discontinuities, or other anomalies. The operator will note possible objects and will "save/update" the volumetric data representation to disk.

The operator will continue this plan/run/inspect procedure until the mapping data is satisfactory. Recovery from system failure will entail rebuilding the volumetric data representation from the raw data point backup.

B.3.1.5 Object Modeling

To create a geometric object to represent an object perceived in the work space, the operator will define a Region Of Interest (ROI), synthesize a geometric object to encompass it, and prepare a textual description of the object.

To establish the ROI, the operator will use multiple arbitrary and/or orthogonal views of the local area. These views will have XYZ scales attached and will display pseudo-coloring based on the property selected by the operator. The operator will define the boundary of the ROI using one or more pairs of cut planes together with the translation, rotation, and scaling controls for the view.

Geometric objects can be synthesized as a polyhedra or from geometric templates. To create a geometric object as a polyhedron, the operator will use the space ball/mouse to draw a 2D polygon on one of the views. The display will provide some dimensional "hints" such as tick marks or measurement cursors. If appropriate, the finished version of the 2D polygon can be added to the library of templates. To create a 3D geometric object from the 2D polygon, the operator will draw a sweep path on one or more of the views. The volume thus swept out can be displayed and the system can show/compute the difference between the swept volume and the enclosed volumetric data. The operator can modify the 2D polygon and/or the sweep paths. S/he can recompute the difference, repeating this sequence until satisfied with the congruence between the geometric object and the volumetric data representing the perceived object

A geometric object can also be synthesized from 2D and/or 3D templates. The operator selects a template item from the library and manipulates it by scaling, translation and/or rotation. The resulting geometric object can be displayed and compared with the volumetric data. The operator can edit the features until the geometric object is satisfactory.

The third modeling activity is the preparation of text data to attach to the geometric object. This data will include the operator's ID, a description of the perceived object and any pertinent observations concerning the perceived object.

The modeling activity is completed when the geometric object and its associated text data have been properly stored in the system. It may also be necessary to convert the geometric object into a (robot compatible) world model format for export to some other system.

B.3.1.6 Comparison with Previous Map Data

There are certain tanks whose waste surface appears to change. A succession of surface maps is desired to quantify such surface movement.

To provide such data, the following steps are needed. The equipment must be installed and the tank mapped (again) in the tank to be mapped. Using a copy of the previous octree, the task space can be defined by setting the region(s) of interest to an "Unknown" state. The operator can then proceed to map these regions, using the procedure described in Sec. 3.1.3. The system can generate 2.5D surface maps to describe the absolute position of the waste surface as well as the change in that position since the last measurement. A gray scale plot of the difference map can be generated and the difference volume can be computed.

B.3.1.7 Remediation Data Archival

Throughout the remediation process, the operations performed and the condition of the tank, as characterized by its volumetric description and other sensor data, will be periodically stored in an archival system to provide an historical record. When periodic archival is required, the operator will repeat the mapping and object modeling activities described in the previous sections in order to obtain an "as-is" description of the tank.

B.3.2 Buried Waste Retrieval

Production facilities generated contaminated waste in the form of cutting fluids (with entrained chips), tools, plastic containers, protective clothing, etc. These materials were placed in 55 gallon steel drums or wooden boxes and buried in large pits which were subsequently filled in and covered over. Now, several decades later, it is necessary to unearth this material and remove it for treatment, repackaging, and safe storage. Historical records of the site contents are not always available or accurate.

This application is very similar to the removal of waste from the underground storage tanks. However, it expands the requirements in several appropriate ways which are included in the following description.

B.3.2.1 Overview

A typical pit containing buried waste is 200 ft wide, 500 ft long and 30 ft deep. Current plans call for a 200 ft long dig face which would move down the 500 ft length of the pit. Remote excavators would be used to remove scoops of earth and place them on a conveyor system for removal to a facility which would decontaminate the soil. Buried objects such as random and orderly piles of 55 gallon drums and partially decayed cartons and wooden boxes will be removed and treated separately. There may be other large objects such as machine tools or contaminated vehicles. Because of the presence of transuranic elements and other radioactive materials, this work must be done remotely. The dig site will be enclosed to contain any volatile materials which may be uncovered.

In some cases, the retrieval process will not be to an entire buried waste site, but will involve a surgical retrieval from a specific region within the site. In such cases, the following description would be tailored accordingly.

The operator and equipment connected with the ICERVS are assumed to be the same as for the underground tank application. Other assumption include the following.

- There is a geographic database for the entire work space. i.e., the entire pit
- The remote vehicle's position is accurately known in the work space.
- The vehicle's robotic system provides accurate information about the position and attitude of the mapping sensors.
- With this information, the task space (that space encompassed by the remote viewing system) can be accurately located within the work space.
- The mapping sensors are line of sight sensors, although not necessarily based on structured light.

In addition, there may be survey data on the material properties of the buried waste, to include temperature, conductivity, chemical composition, radiation levels, etc. This data should be available to the operator during the excavation. Any property measurements made during the excavation should be used to update the survey data.

B.3.2.2 Installation

In preparation, the existing site data will be reviewed and used to develop a mission plan. The physical installation at the site involves obtaining permits, selecting and verifying the retrieval equipment (including mobile platforms), and establishing the necessary power, guidance, and communication links. The logical elements of installation include constructing an initial volumetric data base from the geographic and other existing data, and configuring the ICERVS for the specific sensors to be used at the site.

B.3.2.3 Operation

The operator will have a graphic display of the dig face surface, to include defined objects and any robotic elements present in the task space. The display will be the equivalent of placing a video camera in the workspace. The viewpoint can reflect a real camera position or any other operator-selected position. The operator will use the robotic elements to dig up the earth and place it on the conveyor system. The robotic elements will also pick up and remove small objects such as 55 gallon barrels. Larger objects will be uncovered and left for removal by special excavators. Models of such large objects are needed in the work space to provide guidance to the various remote vehicles which might encounter them or be sent to remove them.

During retrieval, the operator will periodically re-map the workspace. This will allow the operator to observe the progress of the mission as well as provide an historical record. Automatic re-mapping by

sensors attached to the robotic elements may be possible. The operator will interactively define additional geometric objects (using templates or polyhedra) to represent new physical objects revealed in the remapping, or edit existing geometric objects based on information derived from the new map. The operator will also delete from the graphic display representations of those objects which have been removed by the robotic equipment. The update of the graphic display will be completed when the operator resolves any inconsistencies presented by the system. These inconsistencies can be detected when the system compares the various geometric objects with the new dig face data and finds gaps between the surface and the geometric object, i.e., the geometric object appears to be suspended in space.

One of the most important and challenging operator tasks will be to predict underlying objects beneath the dig face surface. To support this task, the retrieval equipment will have a variety of environmental sensors measuring properties such as intrinsic radiation, neutron-activated radiation, ground penetrating radar signature, electrical conductivity, and magnetic signature. To interpret the sensor data, the operator will perform various data pre-processing, analysis, and data visualization activities. S/he will look for patterns in individual sensor signals and multiple-sensor combinations. S/he may also call up and view "reference" information from previous and/or simulated retrieval missions in order to reach an improved understanding of the present dig face. The operator predictions of underlying objects are used to plan subsequent excavation tools and paths. These predictions are continually reviewed and updated as the excavation proceeds and new sensor data is obtained.

When the desired extent of excavation has been accomplished at this task space, the remote vehicle will move to the next (adjacent?) task space. The system will check to see if the current task space has been remapped and if the world model has been updated. In certain modes of operation TBD, the operator will be allowed to proceed to the next task space without a complete update of the current task space.

B.3.3 Production Facility D&D

Many of the facilities which produced the enriched materials are no longer useful and will be decontaminated, dismantled and removed. The disassembly of the process equipment must be done remotely because of the accumulations of radioactive materials at various places in the process flow. Piping elbows and joints are examples of such places. The floor plans and piping diagrams for these plants are old and probably outdated, but serve as initial guidance for remote vehicles which will enter these facilities to disassemble them for subsequent processing of the components. The work space will be taken as the entire plant and the task space as the plant volume within reach of the robotic unit in its present position.

B.3.3.1 Overview

Having arrived at a selected task space, the operator will map the volume and identify the piping elements to be removed. A radiation sensor will be used to scan the pipes to identify regions which are "hot". These places will be labeled and cut planes defined to permit their removal for placement in special containers. The remaining piping elements will then be removed and placed in another receptacle for processing. When all the piping elements have been removed, the remote unit moves to the next task space.

The operator and equipment connected with the ICERVS are assumed to be the same as for the underground tank application. Other assumption include the following.

- There is an architectural database for the entire work space., i.e., the entire plant. At a minimum, the original drawings will be scanned, digitized, and stored in computer form.
- The remote vehicle's position is accurately known in the work space.
- The vehicle's robotic system provides accurate information about the position and attitude of the mapping sensors.
- With this information, the task space (that space encompassed by the remote viewing system) can be accurately located in the work space.
- The mapping sensors are line of sight sensors, although not necessarily based on structured light.

B.3.3.2 Operation

The operator will view the current task space and create geometric objects to describe the structural features of interest. To assists in understanding the video images from the task space, the operator will be able to bring up facility drawings for that task space. He may overlay features on the drawing display to indicate the robot location and other features of interest. He may also make "corrections" to the drawings based on observations in the task space.

To identify objects for dismantlement, the operator will use a real-time display for a selected video camera, either using a standalone monitor or a window in the computer display. The operator will select (e.g., mouse-click) on the video display to enter geometric objects to represent, for example, straight-line pipe segments. These geometric objects will be displayed in wire frame over the video image. The operator can manipulate each geometric object to "fit" the image via a control panel with buttons for scaling, translation, and rotation. The operator will then select three points on each geometric object and command the system to measure the XYZ coordinates using photogrammetry, laser range finder, or structured light, etc. From this data, the system will establish the 3D orientation of the physical object.

Using the geometric objects, the robotic unit will carry out an automatic scan of radiation level. The data can be used to provide a pseudo-color shaded polygon display to assist in planning the disassembly. Using threshold values to determine what is "hot", the computer can suggest cut locations and the operator will confirm/revise and also select grip locations. The selected geometric objects and disassembly information are then uploaded to the robot controller. The disassembly will then be carried out by the robot controller, which will compute paths (trajectories), run these paths, and carry out the disposal actions.

THIS PAGE INTENTIONALLY BLANK

Appendix C: Phase II Revisions To ICERVS System Requirements

The requirements for the ICERVS were first developed from the set of mission profiles set forth in the ICERVS Phase I System Design Report. In determining the Phase I subsystem design, these requirements were studied and a set of derived requirements was determined. This study led to greater insights into the requirements and resulted in the revisions described in Appendix A of the ICERVS Phase I Subsystem Design Report.

During the Phase II design, revisions to the requirements have been made once again to reflect the experience gained in Phase I. In addition, changes in the Mission Profiles (Appendix A) and the inclusion of the buried waste task have necessitated the addition of several new requirements.

C.1. Clarification of Wording

The following requirements have been revised as follows. The changes are indicated in italics.

Delete

- R1.02 **octree representation, property data** -- Each volume element within the octree representation will *optionally include at least 16 bits to represent property information.*

Insert

- R1.02 **octree representation, property data** -- Each volume element within the octree representation will *include the capability for storage of one or more material properties. A material property value may be composed of a single value or an array of single values.*

Phase II will only support single-valued properties.

Delete

- R1.06 **geometric models, geometric primitives** -- The system shall provide geometric primitives for building *models*. These primitives shall include cones, planes, cylinders, spheres, *rectangular solids*, at a minimum.

Insert

- R1.06 **geometric models, geometric primitives** -- The system shall provide geometric primitives for building *geometric objects*. These primitives shall include cones, planes, cylinders, spheres, *rectangular parallelepipeds, and points*, at a minimum.

Delete

- R3.04 **geometric objects: select *model object* and display associated text data** -- The system shall provide the capability to select a *model object* and display the associated textual information.

Insert

- R3.04 **geometric objects: select *geometric object* and display associated text data** -- The system shall provide the capability to select a geometric object and display the associated textual information.

Delete

- R3.05 **geometric objects: shaded or wire frame polygons** -- The system shall provide the capability to display the *model objects* as either wire frame display or as shaded polygons.

Insert

- R3.05 **geometric objects: shaded or wire frame polygons** -- The system shall provide the capability to display the *geometric objects* as either wire frame display or as shaded polygons.

Delete

- R3.06 **octree representation: *update* as input points received and converted** -- The system shall provide the capability for *displaying the volume and updating* the display as input data is added to the octree.

Insert

- R3.06 **octree representation: *update the display* as input points received and converted** -- The system shall provide the capability for *updating* the display as input data is added to the octree.

Delete

- R3.08 **property coloring of *model objects*** - The system shall have the capability to assign colors to a *model object* based on *the property* of the object

Insert

- R3.08 **coloring of *geometric objects by category*** - The system shall have the capability to assign colors to a *geometric object* based on *a selected category* of the object

Delete

- R3.09 **text display of current window parameters** - The system shall provide a textual display of the current view display parameters. These will include cut plane positions, scale factors, *origin of display window*, at a minimum.

Insert

- R3.09 **text display of current window parameters** - The system shall provide a textual display of the current view display parameters. These will include cut plane positions, scale factors, *viewpoint coordinates*, at a minimum.

Delete

- R4.02 **wire frame display over live video** - The system shall provide the capability to display the wire frame *model objects* on a live video image.

Insert

- R4.02 **wire frame display over live video** - The system shall provide the capability to display the wire frame *geometric objects* on a live video image.

Delete

- R5.03 **operator delete *objects*** -- The system shall provide the capability for the operator to delete *model objects*.

Insert

- R5.03 **operator delete *geometric objects*** -- The system shall provide the capability for the operator to delete *geometric objects*.

Delete

- R5.08 **compute *absolute and difference* 2.5D surface maps** -- The system shall provide the capability to generate *absolute and difference* 2.5D surface maps.

Insert

- R5.08 **compute *difference* 2.5D surface maps** -- The system shall provide the capability to generate *difference* 2.5D surface maps.

Delete

- R6.02 **save/retrieve *models* to/from disk** -- The system shall provide the capability for the operator to save *all models* to disk and retrieve *the models* from disk. The system shall be able to support multiple *data libraries* on the same disk. *Only one data library* is active at a time.

Insert

- R6.02 **save/retrieve *waste site datasets* to/from disk** -- The system shall provide the capability for the operator to save *to disk* and retrieve *from disk* the *spatial, geometric, sensor, and property data* for a waste site.

Delete

- R7.01 **input: *x,y,z* position** -- The input data provided by the mapping system or its equivalent will contain full 3D coordinates such as rectangular coordinates (X, Y, Z) or *polar coordinates* (R, j,q).

Insert

- R7.01 **input: *x,y,z* position** -- The input data provided by the mapping system or its equivalent will contain full 3D coordinates such as rectangular coordinates (X, Y, Z) or *spherical coordinates* (R, phi, theta).

Delete

- R7.06 **output: *geometric models, text reports, robot-compatible models*** -- The system shall provide the capability to output the *geometric models* as a text report and in a form suitable for importing into other systems.

Insert

- R7.06 output: geometric models, text reports, robot-compatible models** – The system shall provide the capability to output the *geometric objects, including user defined geometric objects and the geometric object that represents the waste surface*, as a text report and in a form suitable for importing into other systems.

Phase II is limited to user defined objects. Phase III will include the automatic generation and output of the waste surface geometric object.

Delete

- R9.10 surface mapping sensor** - The system shall include non-contacting means for measuring the location of *exposed surfaces* as described for the specific environment of the site. The sensing means shall function reliably in the site environment.

Insert

- R9.10 surface mapping sensor** - The system shall include non-contacting means for measuring the location of *exposed vertical and horizontal surfaces* as described for the specific environment of the site. The sensing means shall function reliably in the site environment.

C.2. Numbering Change

The following requirements have been renumbered to place the requirement in a more appropriate category:

Delete

- R9.08 flexible color TV** -- The system shall provide the operator with color TV images of all exposed surfaces whose view is consistent with the available camera locations at that site. The operator shall be able to selectively explore smaller regions at higher magnification.

Insert

- R4.04 flexible color TV** -- The system shall provide the operator with color TV images of all exposed surfaces whose view is consistent with the available camera locations at that site. The operator shall be able to selectively explore smaller regions at higher magnification.

C.3. New Requirements

The following new requirements have been added:

R1.10 Octree representation: sensor data

Each volume element with the octree representation will optionally allow storage of a sensor dataset. A sensor dataset consists of the location, orientation, other attributes of a sensor together with a response set.

The sensor response set may employ any data format and should not be interpreted by the octree. Possible data types include single values (temperature, pressure, density), vectors (line of sight, wave forms, simple spectra), matrices (2D spectra, video images, radar images). Phase II will support only single-valued sensors.

R1.11 Octree representation: property data interpolation

The system shall provide the capability to do property data interpolation.

Sparse material property data can be manually entered and the system will interpolate between these data points to generate a denser set of data points representing the property data. Alternatively, a region can be selected in a view and the system will update the display and/or tree to fill any surface discontinuities in that region

R3.14 display views of spatial and property data

The system shall provide the capability for generating views of the spatial and property data in the task space.

Phase I and Phase II will provide orthographic views only. Phase III will provide perspective views in addition to the orthographic views. Phase I will provide orthogonal views. Phase II will provide orthogonal and arbitrary views.

R4.03 Real-time CCV

The system shall provide means for the operator to remotely view the remediation task space. The CCV shall be capable of displaying current conditions of the task space.

For Phase II, the camera on the mapping sensor may be used.

R5.09 surface connectivity

The system shall provide the capability for performing connectivity analyses.

Connectivity analysis will consist of identifying all volumetric data elements that are of the same occupancy state (e.g., empty, full, unknown) and spatially connected to a selected (seed) point. The system will be able to, at a minimum, enable-disable the display of the data elements that are not connected, and change the state of all data elements not connected within a selected region.

R5.10 tools for registering sensor data

The system shall provide the capability for

1. Conversion to common units
2. Alignment (rotation, translation) of data sets
3. Compensation for non-linear effects
4. Estimation of resolution and accuracy of individual data points

MTI will investigate registration techniques at the National Laboratories (Idaho, Pacific Northwest, and/or Oak Ridge) for possible inclusion into the Phase II system.

R5.11 combine multiple property data

The system shall provide the capability to combine multiple property data via Boolean operations. As a minimum, union and intersection will be supported.

R6.07 establish an Application Data Structure for storage of waste site datasets

Major ICERVS software components can be configured to utilize either local or remote disk space for storage of application datasets (spatial, geometric, sensor and property data for a waste site). During Phase II, ICERVS will implement the Application Data Structure only on the application (client) platform.

R6.08 support multiple active datasets per waste site

The system shall be able to support multiple datasets for a waste site on the same disk. Each dataset represents a snapshot of the waste site at some definable condition (date/time, depth, etc.). The dataset will include the spatial, geometric, sensor, property data, and saved views associated with the waste site at the time of the snapshot. Multiple datasets for a site may be active at one time.

This capability will allow operators to make on screen visual comparisons between two or more snapshots of a waste site.

R6.09 support multiple active waste sites

ICERVS will support multiple waste sites on the same disk. Examples of waste sites are a single underground storage tank, all or part of a buried waste pit, one or more cells in a production facility, etc. Views of datasets for multiple waste sites may be active at one time.

This capability will allow operators to visually compare datasets from two or more waste sites.

R7.08 Input: sensor data

Sensor data consists of the location, orientation, other attributes, and a response set. The other attributes may pertain to calibration, registration, pre-processing, or other sensor data specific operations. Sensor data is generally processed to yield one or more property data values.

ICERVS will provide an interface to a structured light sensor. Both a control and a data interface will be provided in accordance with the Common Interface For Sensors specification.

ICERVS will provide an interface to the laser range finder sensor. For Phase II, the interface is limited to a data interface. If suitable data is not available, the data interface will be combined with a suitable simulator to produce appropriate data.

ICERVS will provide an interface to the sensor MiniLab. For Phase II, the interface is limited to a data interface. If suitable data is not available, the data interface will be combined with a suitable simulator to produce appropriate data.

A manual means to enter sensor data will probably be part of the laser range finder and/or MiniLab interfaces during Phase II. It may be a useful capability for ICERVS to support a file-based input of sensor data.

R8.02 provide an operator help facility

The ICERVS software shall incorporate an on-line help facility to provide operator assistance.

Minimal help will be provided in Phase I and Phase II. Phase III will incorporate a more extensive help facility.

R8.03 provide hard copy output

The system will provide hard copy output of menu screens, graphics screen, and other user interface windows.

R9.09 operate sensor subsystems remotely from ICERVS computing platform

The system will support the remote operation of attached sensors from the ICERVS computing platform. Remote operation will be accomplished via an ICERVS Common Interface for Sensors. This interface will be based upon a client-server relationship between the computing platform software and the sensor subsystem. Communications will be via RPC and UNIX sockets.

Consideration is being given to the Sandia ISOE and GENISAS packages as the basis of the client-server and the communications interfaces. Issues relating to GISC compatibility should be explored.

THIS PAGE INTENTIONALLY BLANK

Appendix D: Revised System Requirements

This appendix describes all of the ICERVS system requirements that were derived from the mission profiles. The requirements listed below incorporate all revisions from Phase I and Phase II.

(R1) DATA REPRESENTATION

(R1.01) octree representation, spatial data

Spatial data will be represented in octree format and will be provided in rectangular coordinates. Each volume element will represent a state of empty, partial, full, or unknown.

(R1.02) octree representation, property data

Each volume element within the octree representation will include the capability for storage of one or more material properties. A material property value may be composed of a single value or an array of single values.

Phase II will support only single-valued properties.

(R1.03) octree representation, spatial interpolation

The system shall provide the capability to do spatial interpolation.

Sparse data can be manually entered and the system will interpolate between these data points to generate a denser set of data points representing of the spatial data. Alternatively, a region can be selected in a view and the system will update the display and/or tree to fill any surface discontinuities in that region

(R1.04) octree representation, linear resolution of 1:512

The octree shall have a spatial resolution of 1 part in 512, at a minimum, and be expandable.

The 1 part in 512 gives a 9 level tree. For phase I, the input data will have a default resolution of 1:512. The tree structure must have the capability for increased resolution over the whole volume and over portions of the volume.

(R1.05) geometric models, polyhedral objects

The system will provide a representation of polyhedral objects to model real objects in the work volume.

(R1.06) geometric models, geometric primitives

The system shall provide primitives for building geometric objects. These primitives shall include cones, planes, cylinders, spheres, rectangular parallelepipeds, and points at a minimum.

Geometric primitives are analytic geometric shapes.

(R1.07) geometric models, associated text each object

Each geometric object shall have associated textual information. This information shall consist of an object name and object description, at a minimum.

Other attributes may be added to the object data base as the need arises. Possible other attributes are object category, object disposition, radiology, etc.

(R1.08) geometric models, support at least 100 objects, expansion capability

The system shall support an object data base capable of holding at least 100 objects and be expandable to hold more objects.

The ideal data base would only be limited by available storage.

(R1.09) geometric models, enter architectural plans and robot descriptions

The system shall be capable of accepting CAD drawings, architectural plans and models of the robot manipulator(s).

The architectural plans can be entered manually as objects or ideally imported from some CAD system. The architectural plans provide a representation of the physical environment. The robot model will identify the space occupied by other equipment so that it can be avoided by other equipment in the work space.

(R1.10) Octree representation: sensor data

Each volume element with the octree representation will optionally allow storage of a sensor dataset. A sensor dataset consists of the location, orientation, other attributes of a sensor together with a response set.

The sensor response set may employ any data format and should not be interpreted by the octree. Possible data types include single values (temperature, pressure, density), vectors (line of sight, wave forms, simple spectra), matrices (2D spectra, video images, radar images). Phase II will support only singled- valued sensors.

(R1.11) Octree representation: property data interpolation

The system shall provide the capability to do property data interpolation.

Sparse material property data can be manually entered and the system will interpolate between these data points to generate a denser set of the property data values. Alternatively, a region can be selected in a view and the system will update the display and/or tree to fill any surface discontinuities in that region

(R2) OBJECT MODELING

(R2.01) create/modify primitives/templates

The system will support primitives and templates for object modeling. The operator will be able to create and modify geometric templates.

A primitive is an atomic geometric construct. A template is a collection of primitives and/or other templates. A template is a predefined geometric object that the user can use and modify rather than have to build from points and edges each time.

(R2.02) standard templates including station, registration target, core sample

The system shall supply a collection of predefined geometric templates to aid the operator in modeling an object. These templates will consist of simple geometric shapes and single point location markers.

There will be system supplied geometric primitives that the user can use to build the models. These primitives may consist of simple geometric shapes such as rectangles and circles. Other symbols may be supplied to indicate and locate other points of interest.

(R2.03) user can define templates and add to library

The system shall provide the capability for the user to add user defined templates to the library and to delete templates from the library.

This will allow the operator to save frequently used objects in the library for increased productivity. Separate template libraries for each operator and/or site may be a useful feature of the system.

(R2.04) automatic waste surface modeling

The system shall provide the ability to automatically generate a model for the waste surface. This model should only include the surfaces that are not already modeled by other objects.

The waste surface geometric model (polygon representation) is needed by the robot simulator so that it knows where to excavate. This automatically generated model should be easily deleted and regenerated as the waste material is removed. This is a productivity tool.

(R2.05) define/modify/display 2D polygons (sequence of vertices)

The system will allow the operator to define, modify, erase, and display 2D polygons. It is assumed that convex polygons will be sufficient to support all mission profiles.

(R2.06) create/modify/display 3D polyhedral objects (swept volume)

The system shall allow the operator to create, modify, delete and display 3D polyhedral objects. These objects will consist of 2D polygons swept along a path to create the 3D object.

The Phase I system will be limited to straight line sweep paths parallel to one of the coordinate axes. Phase II will allow other straight line sweep paths.

(R2.07) dimensioning tools, e.g., tick marks, measuring cursors

The system shall supply dimensioning tools to aid the operator in creating objects of a specific size and measuring items in the volume. These tools may consist of scales attached to the border of the display windows with tick marks that track the cursor position, and informational dialogs with the numerical information displayed.

(R2.08) attach text to objects

The system shall provide the capability for the operator to attach text to objects.

This text feature should be used to describe the object. The text storage feature should look like a data base with some specific fields for fixed object attributes and a free form description field. There should be a name field which can be optionally displayed in the view window.

(R3) COMPUTER GRAPHICS DISPLAY

(R3.01) translate and scale

The ICERVS will allow the user to scale the display of the volume in the window, from the whole volume to a single volume element. The ICERVS will allow the user to display any portion of the volume in the window.

(R3.02) display coordinate axes

The graphical display shall display the coordinate axes.

Format and content of display TBD

(R3.03) pair of parallel cut planes

The graphical display will support at least a single pair of parallel cut planes to be used in defining the region of the volume to be viewed.

For the Phase I system, these cut planes will be parallel to one of the coordinate system planes. They will be generalized in subsequent phases.

(R3.04) geometric objects: select geometric object and display associated text data

The system shall provide the capability to select a geometric object and display the associated textual information.

(R3.05) geometric objects: shaded or wire frame polygons

The system shall provide the capability to display the geometric objects as either wire frame display or as shaded polygons.

The Phase I system will only implement wire frame display

(R3.06) octree representation: update display as input points received and converted

The system shall provide the capability for displaying updating the display as input data are added to the octree.

There should be an option to have the display updated every N points added to the tree.

(R3.07) octree representation: pseudo-color by property or dimension

The system shall provide the capability to generate a display of an octree using color values to represent dimensional or property information.

The pseudo-color displays can be generated from values in the property tag to provide a visualization of temperature (for example). The dimensional information is derived from a specific point of reference. The display can be colored to distinguish between regions of different resolution data.

(R3.08) Coloring of geometric objects by category

The system shall have the capability to assign colors to a geometric object based on a selected category of the geometric object

This is used to easily distinguish between different categories of objects, for example waste surface, tank structures, and waste objects embedded in the waste surface.

(R3.09) text display of current window parameters

The system shall provide a textual display of the current view display parameters. These will include cut plane positions, scale factors, viewpoint coordinates, at a minimum.

(R3.10) save/recall view parameter set

The system shall provide the ability for the user to save and recall the display window parameters so that a particular view of the volume can be easily reconstructed at a later time.

These view presets shall be saved to disk so that they can be accessed at a later time. Presets should have a name associated and be displayed in a pop-up menu or list.

(R3.11) multiple windows for display of same representation with different parameters

The system shall provide the ability to display multiple views of the same octree representation with independent display parameters.

(R3.12) set view parameters to track sensor station viewpoint

The system shall provide the ability to have the view point of a single display track with the position of an external sensor and/or camera. The point of view can be from any arbitrary location.

This will allow the operator to steer a sensor (camera) to a particular orientation and have the octree model turn to follow the sensor and give an equivalent display of the volume. Phase II will not support perspective display.

(R3.13) display 2.5D surface map data, pseudo-color by Z

The system shall provide the ability to generate a pseudo colored display of the surface. The surface will be in the XY plane and the color will be assigned depending on the distance from a specified XY cut plane and the first occurrence of the surface while moving along the Z axis.

The waste surface only should be considered for this operation, not objects.

(R3.14) display views of spatial and property data

The system shall provide the capability for generating views of the spatial and property data in the task space.

Phase I will display orthogonal views only. Phases II and III will display both orthogonal and arbitrary views. Phase I and Phase II will provide orthographic views only. Phase III will provide perspective views in addition to the orthographic views.

(R4) VIDEO DISPLAY

(R4.01) one monitor per camera plus sensor processing monitor if applicable

The system shall support one video (TV) monitor per video camera in the work volume plus one monitor to display the output of the image processing operations.

(R4.02) wire frame display over live video

The system shall provide the capability to display the wire frame geometric objects on a live video image.

This display will be on the monitor that is used to display the output of the image processing operations.

(R4.03) Real-time CCV

The system shall provide means for the operator to remotely view the remediation task space. The CCV shall be capable of displaying current conditions of the task space.

For Phase II, the camera on the mapping sensor may be used

(R4.04) flexible color TV

The system shall provide the operator with color TV images of all exposed surfaces whose view is consistent with the available camera locations at that site. The operator shall be able to selectively explore smaller regions at higher magnification.

(R5) MANIPULATION AND ANALYSIS

(R5.01) copy octree

The system shall provide the capability to generate a new octree from an existing octree.

As a natural extension of this operation, the operator will be able to create a new octree that is totally unknown.

(R5.02) "set" region within octree to selected state

The system shall provide the capability to set a specific region within the work volume to a specific state.

This allows the operator to "erase" particular areas within the volume. The system can also use this capability to erase a region before re-mapping it. The regions must be a rectangular solid but are not limited to being cubic.

(R5.03) operator delete geometric objects

The system shall provide the capability for the operator to delete geometric objects.

This would be used after an identified object has been removed from the work volume.

(R5.04) scan model for consistency (no suspended geometric objects), issue warning

The system shall provide the capability for comparing the geometric object model and the octree to determine relations and inconsistencies.

These inconsistencies may be as a result of excavating to reveal more of an already mapped geometric object. There would be a gap between the geometric object and the waste surface.

(R5.05) compare volumetric difference between octree and a geometric object

The system shall provide the capability to compute and display the volumetric difference between an octree solid model and the geometric object model.

The display feature will display the octree regions which are inside or outside of a selected model object. The calculation feature will display a table of volume which is enclosed by the object which the octree has identified as empty, partial, full and unknown. The total volume of the model object will also be displayed.

(R5.06) compute volumetric difference between 2 octrees

The system shall provide the capability to compute and display the volumetric difference between an octree solid model and another octree solid model.

The display feature will display the union, intersection, or subtraction etc. of the two trees in different colors. The calculate feature will display the volumes of these regions.

(R5.07) compute 2.5D surface map as projection of 3D data

The system shall provide the capability to generate 2.5D surface maps as a projection of 3D data on a user specified plane, using a user selectable, uniform grid spacing.

(R5.08) compute difference 2.5D surface maps

The system shall provide the capability to generate difference 2.5D surface maps.

The system can compute the difference in volume between two surface maps to determine how much material has been removed or if any subsurface bubbles are forming.

(R5.09) surface connectivity

The system shall provide the capability for performing connectivity analyses.

Connectivity analysis will consist of identifying all volumetric data elements that are of the same occupancy state (e.g., empty, full, unknown) and spatially connected to a selected (seed) point. The system will be able to, at a minimum, enable-disable the display of the data elements that are not connected, and change the state of all data elements not connected within a selected region.

(R5.10) tools for registering sensor data

The system shall provide the capability for

1. Conversion to common units
2. Alignment (rotation, translation) of data sets
3. Compensation for non-linear effects
4. Estimation of resolution and accuracy of individual data points

MTI will investigate registration techniques at the National Laboratories (Idaho, Pacific Northwest, and/or Oak Ridge) for possible inclusion into the Phase II system.

(R5.11) combine multiple property data

The system shall provide the capability to combine multiple property data via Boolean operations. As a minimum, union and intersection will be supported.

(R6) MISCELLANEOUS FUNCTIONS

(R6.01) operator edit system parameters

The system shall provide the capability for the operator to edit the system parameters.

(R6.02) save/retrieve waste site data sets to/from disk

The system shall provide the capability for the operator to save to disk and retrieve from disk the spatial, geometric object, property, and sensor data sets for a waste site.

(R6.03) rebuild octree from backup raw data points

The system shall provide the capability to rebuild the octree from raw data points saved from the sensor data stream.

The system should be able to build the octree from a data stream of the correct format. It should not care as to the origin of that data stream. It could be from a sensor, disk file, or user input.

(R6.04) maintain operator log and operator note book for observations or other notes

The system shall provide the capability for the operator to enter observations into an on-line notebook.

The operator will login to the system with a unique identifier (and possibly a password). The operator ID and time stamp will be added to the operator log. The operator ID will be used as an audit tag for model objects and other operations.

There will be a place to enter comments, observations, etc. that pertain to the work volume as a whole, such as type of structure, location, notes about interior conditions, last time accessed, etc.

The form of the notebook is TBD. Questions to be addressed include: is it a simple, large text editing window, or is a data structure required? Are the entries date stamped? Can the operator delete log entries?

(R6.05) support multiple systems of units

The system shall provide the capability for entering and displaying data in a user selected system of units. The system will support feet/inches and meters/mm as a minimum.

The units will be in standard units internally and will be converted at the interfaces.

(R6.06) define disassembly data

The system shall provide the capability for the operator to specify disassembly and disposal options for objects in the work volume.

The system will allow the operator to specify cut and grip locations along with disposal options for D&D applications.

(R6.07) establish an Application Data Structure for storage of waste site datasets

Major ICERVS software components can be configured to utilize either local or remote disk space for storage of application datasets (spatial, geometric, sensor and property data for a waste site). During Phae II, ICERVS will implement the Application Data Structure only on the application (client) platform.

(R6.08) support multiple active datasets per waste site

The system shall be able to support multiple datasets for a waste site on the same disk. Each dataset represents a snapshot of the waste site at some definable condition (date/time, depth, etc.). The dataset will include the spatial, geometric, sensor, property data, and saved views associated with the waste site at the time of the snapshot. Multiple datasets for a site may be active at one time.

This capability will allow operators to make on screen visual comparisons between two or more snapshots of a waste site.

(R6.09) support multiple active waste sites

ICERVS will support multiple waste sites on the same disk. Examples of waste sites are a single underground storage tank, all or part of a buried waste pit, one or more cells in a production facility, etc. Views of datasets for multiple waste sites may be active at one time.

This capability will allow operators to visually compare datasets from two or more waste sites.

(R7) DATA INTERFACE

The input of data to the octree has the following requirements.

(R7.01) input: x,y,z position

The input data provided by the mapping system or its equivalent will contain full 3D coordinates such as rectangular coordinates (X, Y, Z) or spherical coordinates (R, phi, theta).

(R7.02) input: optional resolution

The input data provided by the mapping system or its equivalent may contain the resolution of the data point. If no resolution is specified with the input data, the default resolution will be used.

The form of the resolution specification is TBD.

(R7.03) input: optional property values

The property values for the volume element.

If no value is specified, then a default code indicating no property will be used.

(R7.04) input: optional sensor locations

The input data provided by the mapping system or its equivalent may contain the location of the sensors that measured the data.

Position data are needed for line of sight sensors, and are used for sculpting the octree. For example, if the surface was measured from the specified sensor location, then the volume must be clear between the sensor and the surface.

(R7.05) input: station angles for visual inspection display

The input data provided by the mapping system or its equivalent may contain the sensor station angles so they can be displayed to the operator during the mapping operation.

The angle display may be either by displaying lines of sight on the octree display or by displaying the angles in a text window.

(R7.06) output: geometric models, text reports, robot-compatible models

The system shall provide the capability to output the geometric objects, including user-defined geometric objects and the geometric object that represents the waste surface, as a text report and/or in a form suitable for importing into other systems.

Phase II is limits to user defined geometric objects. Phase III will include the automatic generation and output of the waste surface geometric object.

(R7.07) output: 2.5D surface map, absolute and difference

The system shall provide the capability to output a 2.5D surface map in an absolute version and as a difference between two surfaces.

The format of these maps is TBD.

(R7.08) Input: sensor data

Sensor data consists of the location, orientation, other attributes, and a response set. The other attributes may pertain to calibration, registration, pre-processing, or other sensor data specific operations. Sensor data is generally processed to yield one or more property data values.

ICERVS will provide an interface to a structured light sensor. Both a control interface and a data interface will be provided in accordance with the Common Interface For Sensors specification.

ICERVS will provide an interface to the laser range finder sensor. For Phase II, the interface is limited to a data interface. If suitable data is not available, the data interface will be combined with a suitable simulator to produce appropriate data.

ICERVS will provide an interface to the sensor MiniLab. For Phase II, the interface is limited to a data interface. If suitable data is not available, the data interface will be combined with a suitable simulator to produce appropriate data.

A manual means to enter sensor data will probably be part of the laser range finder and/or MiniLab interfaces during Phase II. It may be a useful capability for ICERVS to support a file-based input of sensor data.

(R8) OPERATOR INTERFACE

(R8.01) graphic tools: spaceball and/or mouse, pull down menus, dialog boxes

The system shall provide the capability to accept input from the operator through either a spaceball or a mouse. The operator shall interact with the system through a graphical interface using pull down menus, dialog boxes, on screen controls, at a minimum.

The intent is to make the system easy to use and to take full advantage of the graphical interface. The Phase I and Phase II systems will use Motif to achieve the goals of the graphical interface.

R8.02 provide an operator help facility

The ICERVS software shall incorporate an on-line help facility to provide operator assistance.

Minimal help will be provided in Phase I and Phase II. Phase III will incorporate a more extensive help facility.

R8.03 provide hard copy output

The system will provide hard copy output of menu screens, graphics screen, and other user interface windows.

(R9) SENSORS

(R9.01) run sensor station positioning using position and rate commands

The system shall provide the capability to run the sensor station positioning equipment via tele-operation using position and rate commands. These commands will come from the operator through either a spaceball or a mouse.

(R9.02) display sensor line-of-sight

The system shall provide the capability to display the station line-of-sight.

The display is envisioned to be as a directed vector superimposed on a view of the volume.

(R9.03) text display of station angles

The system shall provide the capability to display the station positioning angles on the screen.

(R9.04) automatically map surfaces

The system shall provide the capability to automatically map waste surfaces.

This capability will allow the operator to specify areas of the surface to be mapped. The system will convert these area specifications into a list of locations where measurements are to be taken. The purpose here is to increase operator productivity.

(R9.05) automatic surface mapping parameters

The system shall provide the capability for the operator to enter parameters to guide the automatic mapping of the surface. These parameters will include spacing between scan lines, number of data points per scan line (image) and nominal location (i.e. Z height) of the surface, at a minimum.

(R9.06) draw/display scan paths

The system shall provide the capability for the operator to enter the scan paths for the automatic surface mapping to follow by drawing directly on an image of the volume to be mapped. These scan paths will be displayed for operator verification.

Scan paths should be saved so that they can be reused when remapping the site. All scan path parameters should be saved to facilitate remapping.

(R9.07) backup raw data every PARAM points

The system shall provide the capability for saving the data stream from the sensors.

The raw data can be backed up periodically or saved immediately when they are received. The data can be time tagged.

(R9.08) not used

(R9.09) operate sensor subsystems remotely from ICERVS computing platform

The system will support the remote operation of attached sensors from the ICERVS computing platform. Remote operation will be accomplished via an ICERVS Common Interface for Sensors. This interface will be based upon a client-server relationship between the computing platform software and the sensor subsystem. Communications will be via RPC and UNIX sockets.

Consideration is being given to the Sandia ISOE and GENISAS packages as the basis of the client-server and the communications interfaces. Issues relating to GISC compatibility should be explored.

(R9.10) surface mapping sensor

The system shall include non-contacting means for measuring the location of exposed vertical and horizontal surfaces as described for the specific environment of the site. The sensing means shall function reliably in the site environment.

(R9.11) sensor performance

The accuracy and repeatability of measurements made by the surface mapping sensor shall be consistent with the requirements for exploration and excavation in the site environment. The surface mapping sensor shall support the real time control of excavation operations.

(R10) SITE ENVIRONMENT

(R10.1) surface characteristics

The system shall be able to sense surfaces whose characteristics vary widely from site to site. Elements that vary include the following.

- o Reflectivity, color, roughness, and texture (Bare metal, salt crystals, earth, liquids)
- o Size of area to be mapped (Underground storage tanks at 75 ft. in diameter to factory work cells at 15 ft in diameter)
- o Dimensions of features of interest (Barrels at several feet to piping at a few inches.)
- o Orientation of surfaces, both global (walls and ceilings vs. waste surface) and local (objects, clods of earth)

(R10.2) illumination and visibility

The system shall provide the illumination necessary for effective and useful operation of the color TV camera and the surface mapping sensor. The atmosphere at the site environment may be humid or dusty.

(R10.3) environmental constraints

The TV camera and surface mapping sensor shall function properly for a specified period of time in the physical environment of the site. The system design shall reflect adequate consideration for the temperature, humidity, and radiation levels in the environment as well as possible exposure to toxic chemicals. For certain sites, the environment is considered potentially explosive. Possible sensor locations may be limited by, e.g., the available tank penetrations.

(R10.4) design constraints

Equipment intended for use in hazardous environments shall be designed for ease of decontamination. Constraints on the physical size and weight of system components exist at different sites.

There are a limited number of entry ports into underground storage tanks. There can be small (12 inch diameter) openings through which the equipment must pass.

END

**DATE
FILMED**

3/8/94