I. Program Structure and Approach

A. Introduction

The Incident Management & Decision Support System (IMDSS) project delivered a web-enabled software suite to collect, share and process data and events as they occur, providing situation awareness at the local emergency management level and incident management level. The IMDSS delivered near-real-time fusion of data from a variety of local sources providing incident managers with the ability to monitor and respond to emergency situations in a timely fashion.

This project integrated the IMDSS software suite with a mobile tactical operations center called MITOC in order to support the information and processing needs of the on-site incident commander. The MITOC is a man portable communications and processing platform that provides interoperable communications and data processing capability both for semi-permanent fixed use and mobile applications.

B. Program Concept of Operations

1. IMDSS Initial Requirements

   a. Interviewed users and observed training/exercise and/or demonstrations to verify requirements.
      - Defined reporting forms and user interfaces - At the start of the program the proposal team interviewed different users and participated in available exercises and/or demonstrations to better understand the data requirements.
      - Determined what incident management specific data sources are available – The proposal team interviewed users and managers to determine what sensors and data sources were currently available. This effort resulted in a table of potential data sources and their capabilities/issues.
      - Determined incident management data processing requirements – As with data sources, the proposal team interviewed users and managers to determine what types of processing applications were required and how those requirements were being met.
      - Determined user interface requirements – The first step in creating an efficient user interface was to talk to the potential user and understand how things were currently done, and what ideas the user had for improving or automating this process. The proposal team interviewed users and managers as well as participate in available exercises in order to define an initial set of display requirements.
   
   b. Added message generation capability to support sharing and grouping of alerts for IMDSS – The IMDSS automated message generated system was created to support sharing alerts between users and grouping alerts based on user determine criteria. Wherever
possible the proposal team predefined groups of alerts based on anticipated user roles in order to expedite new users.

c. Added IMDSS collaboration server to support documents – The Collaboration server supported the sharing of objects between multiple users. This included maps, layers, and geometries. This server was extended to support the sharing of documents in the form of files. The files could be of any format including, but not limited to, Microsoft Word documents, Adobe PDF files, and text files.

d. Identified existing systems for collaboration – It was highly unlikely that any single system would ever meet the needs of the entire incident management community. It was imperative however that all systems interoperate seamlessly with any other systems in use. The proposal team identified major systems that were in use with incident response for collaboration/data integration.

2. IMDSS Development Efforts

a. Created schemas to properly manage forms input
b. Data Adapters
   • Determined whether existing data adapters supported these data sources and configured those adapters.

c. Implemented new data adapter(s) as required

c. Added additional infrastructure features to support IMDSS needs
   • Configured existing processing applications – The proposal team built processing applications to meet incident management processing needs based on identified requirements.
   • Added alerts to include time based constraints – Frequently; events were only of interest if they occurred at specific times. An example would be the output of a motion detector. During the day when people were at work, a motion detector fired frequently and was not necessarily of interest. However at night, this event was generally more important and should be acted upon immediately.
   • Added features to collaboration server to support documents – The Collaboration server supported the sharing of objects between multiple users. This included maps, layers, and geometries. This server supported the sharing of documents in the form of files. The files could be of any published format including, but not limited to, Microsoft Word documents, Adobe PDF files, and text files.

d. Customized IMDSS displays based on requirements defined during interviews

e. Attended/Participated in exercises and/or demonstrations - Wherever possible, without affecting the original goal of the exercise, the proposal team allowed the user to utilize the system. The system’s performance could then be compared against the exercise outcome to evaluate improvements or issues to identify potential exercises for latter participation.

4. IMDSS Functions and Features Development
a. Continued development of forms based applications
   • Implemented forms using commercially available forms generation technology to manage forms and support dynamic creation of new forms. Continued to customize IMDSS displays based on requirements identified during interviews.

b. Continued to implement new data adapter(s) as required.

c. Continued to extend IMDSS features to support project needs
   • Extended IMDSS alerts to maintain object state – Frequently it was not enough to know whether a particular attribute was above or below a threshold but when it crosses the threshold. This required knowledge of the objects previous state. IMDSS maintained a transaction history, and hence the state, for all objects. The IMDSS alerting mechanism leveraged this transaction.
   • Extended IMDSS alerts to support comparison against known situation – Frequently the status of a given object was only important if other conditions were met. The IMDSS message generation was extended to allow events to be compared against the known situation and determine whether to generate an alert.
   • Extended IMDSS collaboration server to include recipients – When the author/owner of a document shared it, they had an option to add a list of recipients.

5. IMDSS Functions and Features Development

a. Continued spiral development to implement new features
   • Continued to implement forms using commercially available forms generation technology. This process continued to evolve new features and functionality based on needs identified during interviews and exercises. Continued to implement user interface to manage forms and support dynamic creation of new forms.
   • Investigated options for providing additional functionality. Continued to customize IMDSS displays based on requirements defined above.
   • For each system that supported interoperability, continued to design and implement required interfaces.

b. Extended IMDSS message generation to support additional output channels –

c. Added capability to exclude specific alerts for a given user – In order to limit the amount of information a user received, the user should be able to easily exclude particular messages that occurred too frequently without having to redefine his/her event filters. The user interface made it clear that the user had excluded events.

d. Added IMDSS collaboration server feature to use alerting mechanism to notify recipients/author of changes in document status – The collaboration server was extended to use the alerting system described above to alert recipients that a new/updated document is available.
e. Generated Concept of Operations (CONOPS) – A document was generated explaining how IMDSS was intended to be used in both day to day operations as well as during incident response. This document attempted to cover a variety of situations and explain how to best leverage the IMDSS.

6. Task 7 IMDSS Functions and Features Development

a. Continued to implement user interface to manage forms and support dynamic creation of new forms.
b. Continue to investigate options for providing additional functionality – The intent was to incorporate existing processing applications and features as opposed to developing them again. Where existing functionality does not meet the requirements, existing applications or software was evaluated for potential incorporation into the IMDSS architecture. Where existing applications or software was available, the proposal team defined an interface between that software and IMDSS. Since the number of potential processing applications were quite large, these interfaces were implemented based on need to support exercises or demonstrations.
c. Updated CONOPS document.
d. Continued to identify and attend available exercises and/or demonstrations.

7. IMDSS Enhancements and Draft Documentation

a. For systems that did not currently support collaboration, the team generated a report describing potential enhancements required to support collaboration –b. Generated User Documentation – The intent was that the system was as easy to use as possible. For all user visible applications, a detailed help file was available. In addition, a tutorial was developed that describes the overall system, what it does, and how to get started.

8. IMDSS Refinement & Documentation

a. Continued to Generate User Documentation.
b. Continued development to add new features and functionality based on updated user needs.

9. Task 12 IMDSS Final Documentation

a. Generated User Documentation.
b. Identified and attended available exercises and/or demonstrations.
c. Completed product development, marketing plan, and packaging plan.
II. Results and Conclusions

The project delivered a web-enabled software suite to collect, share and process data and events as they occurred during any type of emergency response situation, providing situation awareness at the local emergency management level and incident management level. The IMDSS delivered near-real-time fusion of data from a variety of local sources providing incident managers with the ability to monitor and respond to emergency situations in a timely fashion. The project delivered software at Technology Readiness Level 9 (TRL 9).

This project also provided the integration of the IMDSS software suite and the MITOC hardware platform in order to support the information and processing needs of the onsite incident commander.

III. Transition to Use and Commercialization

The research team worked closely with project commercialization partner, ElanTech, Inc. and Kentucky based emergency response departments and personnel during the development and pilot testing phases. IMDSS was at a Technology Readiness Level (TRL) level 9 at project completion. IMDSS was on track for system test, launch and operation as a commercial product. IMDSS was deployed as a web-based solution paired with the Mobile Telecom and Information Operations Center (MITOC) to provide internet access to first responders operating IMDSS in emergencies where communications infrastructures have been damaged or lost. The immediate target customer for IMDSS was the regional and local emergency manager in rural America. The intent was to provide a low cost, fully functional Incident management tool to first responders in disadvantaged communities. The company was already formed to commercialize the technology, company capitalization efforts were underway, and initial customers have been identified for early deployments. ElanTech, Inc. was working to qualify IMDSS for DHS System Efficacy through Commercialization, Utilization, Relevance and Evaluation (SECURE) Program and DHS Responders Knowledge Base (RKB), an online government database of tools available to support first responders.