



## **Integrating Weather Effects Data Into a Human Computer Interface to Aid the Battlefield Planning Process**

**by Holly Ingham, Timothy Rogers, Mario Torres,  
John Raby, and Donald Hoock**

**ARL-TR-3766**

**April 2006**

## **NOTICES**

### **Disclaimers**

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

# **Army Research Laboratory**

Aberdeen Proving Ground, MD 21005-5067

---

---

**ARL-TR-3766**

**April 2006**

---

## **Integrating Weather Effects Data Into a Human Computer Interface to Aid the Battlefield Planning Process**

**Holly Ingham, Mario Torres, John Raby, and Donald Hoock  
Computational and Information Sciences Directorate, ARL**

**Timothy Rogers  
University of Maryland**

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved</i> OMB No. 0704-0188		
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. <b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b>					
<b>1. REPORT DATE (DD-MM-YYYY)</b> April 2006		<b>2. REPORT TYPE</b> Progress		<b>3. DATES COVERED (From - To)</b> June 2004–June 2005	
<b>4. TITLE AND SUBTITLE</b> Integrating Weather Effects Data Into a Human Computer Interface to Aid the Battlefield Planning Process			<b>5a. CONTRACT NUMBER</b>		
			<b>5b. GRANT NUMBER</b>		
			<b>5c. PROGRAM ELEMENT NUMBER</b>		
<b>6. AUTHOR(S)</b> Holly Ingham, Timothy Rogers, * Mario Torres, John Raby, and Donald Hoock			<b>5d. PROJECT NUMBER</b> P622783Y10		
			<b>5e. TASK NUMBER</b>		
			<b>5f. WORK UNIT NUMBER</b>		
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> U.S. Army Research Laboratory ATTN: AMSRD-ARL-CI-CT Aberdeen Proving Ground, MD 21005-5067			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b> ARL-TR-3766		
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b>			<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>		
			<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>		
<b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited.					
<b>13. SUPPLEMENTARY NOTES</b> *University of Maryland, College Park, MD 20742					
<b>14. ABSTRACT</b> A critical element of battle command decision-making support in the battle space is answering the commanders' priority intelligence requirements (PIRs) in a timely manner and with adequate accuracy. The sheer volume and nature of information reported to analysts and decision makers far exceeds their capability of processing it within the enemy's decision cycle. In 2003, the Army initiated an Army Technology Objective program titled "Fusion Based Knowledge for the Future Force (FBKFF)" to conduct the research and development necessary to provide automated support for answering PIRs at the Unit of Action (brigade-level force). This program is a collaboration between the Intelligence and Information Warfare Directorate of the Communications-Electronics Research, Development, and Engineering Center and the Computational and Information Sciences Directorate (CISD) of the U.S. Army Research Laboratory (ARL). ARL's participation in FBKFF has been focused on obtaining, storing, and assimilating the data necessary to perform the data fusion involved in automated decision support. Personnel from CISD's Computer and Communications Division have teamed with personnel from the CISD Battlefield Environment Division (BED) and the University of Maryland to develop a system to integrate weather forecasts and battlefield weather effects data into an automated decision support tool for intelligence analysts. This report explores the integration development and describes the decision support tool that has been developed to support this and other Army program objectives.					
<b>15. SUBJECT TERMS</b> weather effects data, battlefield planning, decision aid, data proxy					
<b>16. SECURITY CLASSIFICATION OF:</b>			<b>17. LIMITATION OF ABSTRACT</b>  UL	<b>18. NUMBER OF PAGES</b>  24	<b>19a. NAME OF RESPONSIBLE PERSON</b> Holly Ingham
<b>a. REPORT</b> UNCLASSIFIED	<b>b. ABSTRACT</b> UNCLASSIFIED	<b>c. THIS PAGE</b> UNCLASSIFIED			<b>19b. TELEPHONE NUMBER (Include area code)</b> 410-278-9192

---

## Contents

---

<b>List of Figures</b>	<b>iv</b>
<b>1. Introduction</b>	<b>1</b>
<b>2. Integrated Weather Effects Decision Aid Overview</b>	<b>1</b>
<b>3. Decision Support Tool Overview</b>	<b>5</b>
<b>4. University of Maryland’s (UMD) Army Weather Effects (AWE) Data Proxy</b>	<b>7</b>
4.1 AWE Overview .....	7
4.2 IWEDA Database Table Descriptions and Usage .....	9
4.3 IWEDA Ontology.....	9
4.4 AWE Proxy .....	10
4.5 Decision Support Tool Interface With AWE Proxy.....	11
<b>5. Future Directions</b>	<b>15</b>
<b>Distribution List</b>	<b>16</b>

---

## List of Figures

---

Figure 1. Initial IWEDA map interface. ....	2
Figure 2. Initial IWEDA content control. ....	3
Figure 3. Mission component selection. ....	3
Figure 4. Weather effects selection dialog.....	4
Figure 5. Selected weather effects map display.....	4
Figure 6. Grid cell details selection. ....	5
Figure 7. Grid cell weather effect details.....	5
Figure 8. Screen shot of the decision support tool interface.....	6
Figure 9. Architecture of integrated system.....	8
Figure 10. Three base IWEDA graph node extraction queries.....	10
Figure 11. Two base IWEDA graph edge extraction queries.....	10
Figure 12. AWE rendered weather effect overlay grid.....	11
Figure 13. AWE rendered grid cell detail page. ....	12
Figure 14. Three-screen decision support tool interface display. ....	12
Figure 15. Weather controls box.....	13
Figure 16. Mission lists.....	13
Figure 17. Time stamps.....	14
Figure 18. Weather overlay.....	14

---

## **1. Introduction**

---

The current joint vision for 2010 and beyond is a smaller, lighter, more maneuverable force where the reduced lethality and survivability issues are overcome through the operational use of critical information delivered to the right person at the right time. This concept is known as network centric warfare. One potential area of research to enable this concept is in data fusion. The Computational and Information Sciences Directorate (CISD) of the U.S. Army Research Laboratory (ARL) has teamed with the Intelligence and Information Warfare Directorate of the Communications-Electronics Research, Development and Engineering Center on an Army Technology Objective (ATO) program titled “Fusion Based Knowledge for the Future Force (FBKFF)” to conduct the research and development necessary to provide automated support for battle command decision making, in particular answering the commanders’ priority intelligence requirements at the unit of action (brigade-level force).

ARL’s participation is focused on obtaining, storing, and assimilating the data necessary to perform the data fusion involved in automated decision support. This data gathering and filtering process includes the use of ontologies, data schemas, data mining techniques, and database research. Since weather often plays an important role in the decision-making process and weather effects data are readily available through the CISD Battlefield Environment Division (BED) Integrated Weather Effects Decision Aid program (IWEDA), the CISD Computer and Communication Division has collaborated with the CISD BED and the University of Maryland to integrate the weather effects data available through IWEDA into a prototype intelligence analyst’s decision support tool.

---

## **2. Integrated Weather Effects Decision Aid Overview**

---

IWEDA is a rule-based system that is used to generate qualitative mission-relevant weather impact warnings. These warnings are produced when environmental conditions exceed critical levels for mission-relevant platforms, weapon systems, operations, and soldier performance. Each system (platforms, weapons, operations, and soldiers) has its own list of relevant rules that identify the critical meteorological parameters and value thresholds. The rules are used to create a red-amber-green (unfavorable-marginal-favorable) weather impact by evaluating one or a combination of the environmental parameters that affect each system. Results are displayed via a tabulated matrix of impacts vs. time or as red-amber-green impacts spatially displayed as map overlays for the region of interest. IWEDA rules, which interact with the weather database to determine impacts on the selected system(s), are determined from system concepts, military doctrine, and technical manuals and are embodied in a computer database of rules that cross

reference systems, meteorological parameters, and critical value thresholds. These environmental impact rules have been validated through the Training and Doctrine Command's combat development centers and schools, field manuals, technical manuals, subject matter experts, and the National Ground Intelligence Center. IWEDA is currently fielded as part of the Army Battle Command System (ABCS) Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance tactical weather system, the Integrated Meteorological System.<sup>1</sup>

The IWEDA package (written in Java) provides an intuitive graphical user interface (GUI) for selecting and viewing weather effect impacts overlaid on a Joint Mapping Toolkit (JMTK) (comanaged by the Defense Systems Information Agency and the National Geospatial-Intelligence Agency) map interface. While the Java interface ensures cross-platform compatibility and the underlying MySQL data storage provides a powerful, yet inexpensive, data storage solution, the current implementation appears chained to the custom JMTK GUI used in the legacy ABCS, with little data overlay export capability for use with other software applications (such as CISD's decision support tool interface).

The IWEDA software renders a three-window interface through which the user sets the desired mission component and weather effect filters. The user can then explore the weather impacts through selections in the various windows. A quick demo session might appear as follows:

- a. Initialize system (launch MySQL server, IWEDA JChart [figure 1] and IWEDA GUI [figure 2]).

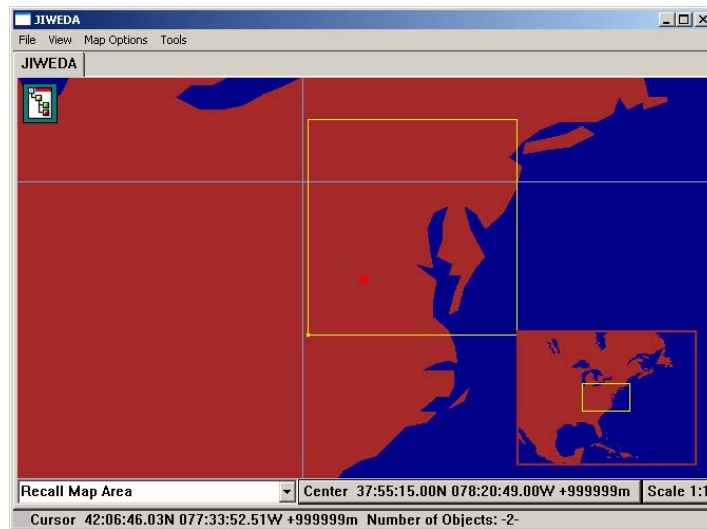


Figure 1. Initial IWEDA map interface.

---

<sup>1</sup>Shirkey, R. C.; Gouveia, M. Weather-Impact Decision Aids: Software to Help Plan Optimal Sensor and System Performance. *CrossTalk: The Journal of Defense Software Engineering* 2002.



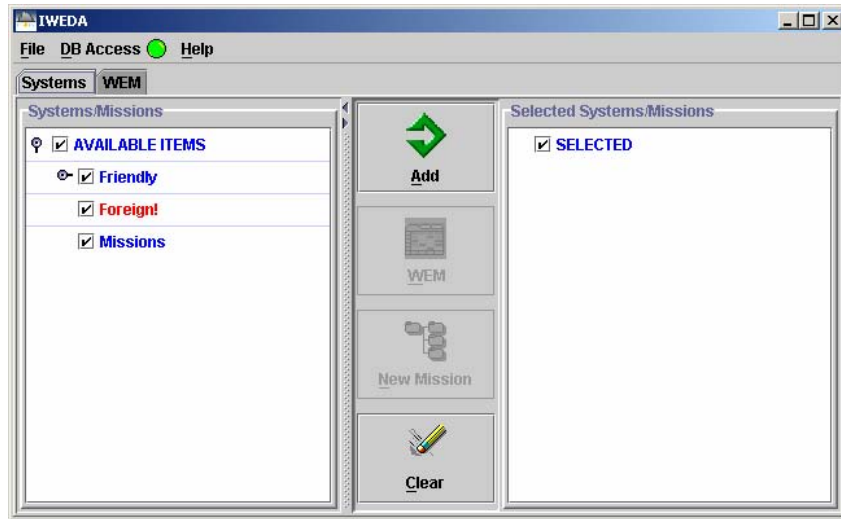


Figure 2. Initial IWEDA content control.

- b. Select and add desired mission components (e.g., A-10, F-15E, M109 Howitzer [see figure 3]) and render weather effects matrix (WEM) dialog (see figure 4) by clicking on the WEM button.

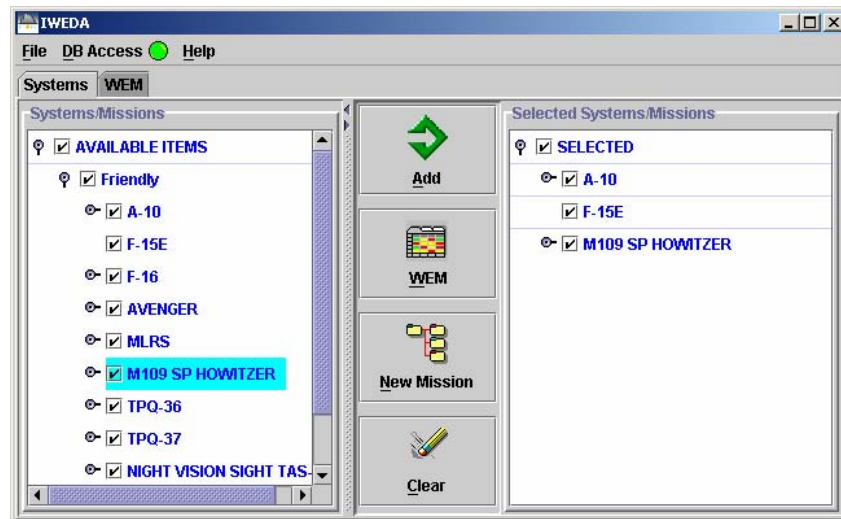


Figure 3. Mission component selection.

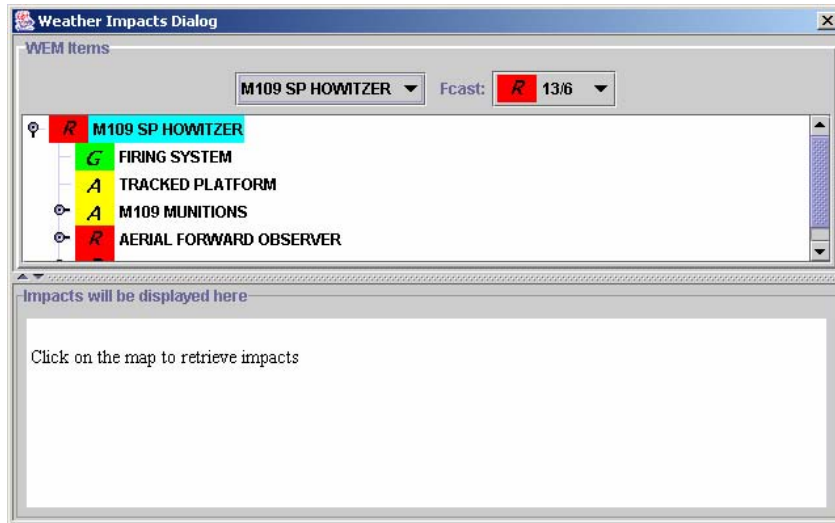


Figure 4. Weather effects selection dialog.

- c. Select an entry in the weather effects dialog tree list (figure 4) and note the JChart map overlay grid display (figure 5).

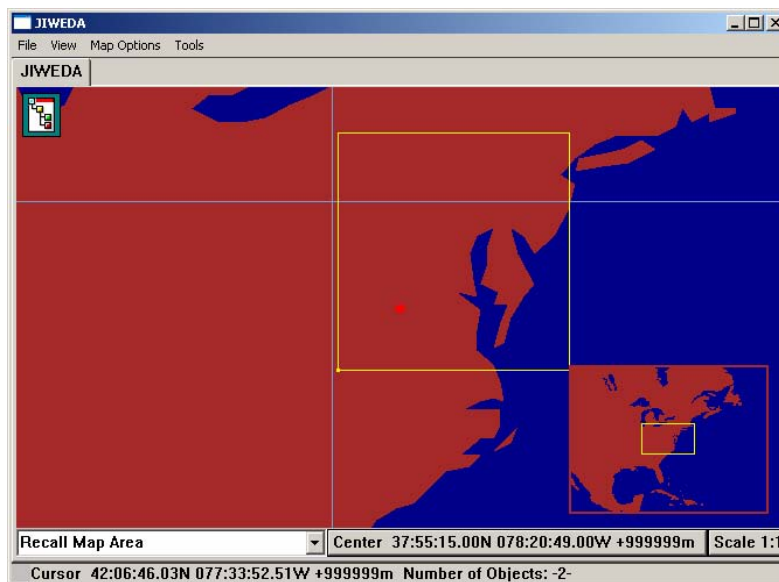


Figure 5. Selected weather effects map display.

- d. Select an interest area grid cell on the JChart map (figure 6) and note the WEM dialog bottom panel update to include the weather impact effects for the various mission components (figure 7).

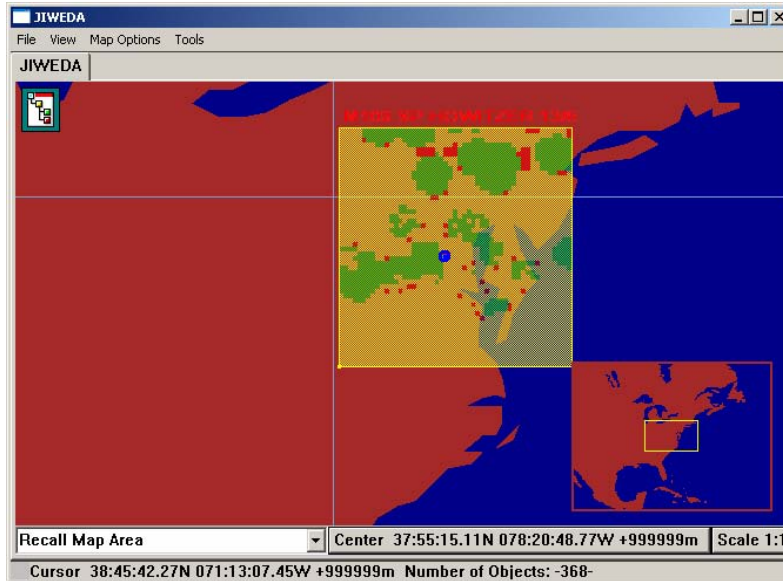


Figure 6. Grid cell details selection.

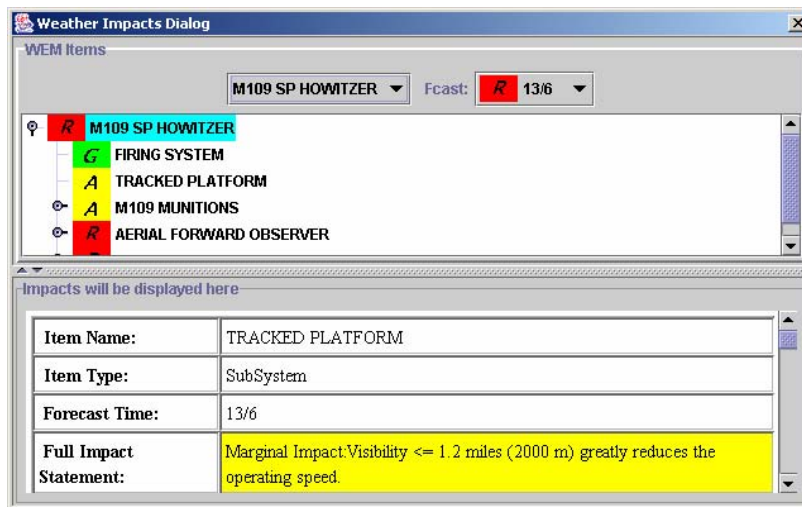


Figure 7. Grid cell weather effect details.

---

### 3. Decision Support Tool Overview

---

ARL has established the Collaborative Technology Alliances (CTA) program, collaboration among government, industry, and university researchers, to achieve affordable transition of innovative technologies. The overall program is composed of five CTA's in advanced sensors, power and energy, advanced decision architectures, communications and networks, and robotics. CISD participates in the advanced decision architectures and has teamed with SA Technologies,

an industry partner, to develop a design strategy for dealing with potential data overload on the battlefield while supporting high levels of situation awareness in military battle command. The result of this collaboration was a prototype decision support interface tool suite for the intelligence officer and the logistics officer. For the purposes of the FBKFF ATO, the intelligence officer tool suite has been further developed to include the weather impact data from the IWEDA program.

The prototype decision support tool interface is a three-screen approach to improving battlefield situation awareness by getting the right information to the right person in a form they can easily assimilate. The decision support tool is just one piece of a larger tool suite designed and targeted for brigade-level staff elements. Each staff element would have a display interface similar to the intelligence officer's decision support tool interface used for this demonstration. Each staff element's decision support interface has tools and overlay options tailored to that staff's information needs. As our battlefield information technology expands, so will the amount of information that will be received. Since the intelligence officer will most likely be receiving a majority of the data from the field, ARL chose to prototype the intelligence officer's decision support tool design concept for proof of concept. The middle screen provides a view of the user defined operating picture (UDOP). This is the common view of the battlefield that all staff elements see. The right screen provides a map of the intelligence officer's view of the battlefield. It shows information obtained through various intelligence assets such as spot reports, sensor input, etc. This map is used by the intelligence officer to aid him in de-conflicting information and deciding what is important to show on the UDOP. Once the intelligence officer has verified and resolved any conflicting reports, the information is sent to the UDOP map. The left screen provides a World Wide Web type of search tool that would include secure sites as appropriate to use during planning. This screen also contains an asset display to allocate and view intelligence assets, a list of the commander's critical information requirements, a chat tool, and an alert tool to alert other staff elements of impending events that could affect their situation. Figure 8 shows a view of the decision support tool.



Figure 8. Screen shot of the decision support tool interface.

---

## **4. University of Maryland's (UMD) Army Weather Effects (AWE) Data Proxy**

---

The University of Maryland's involvement in the FBKFF ATO includes the exploration and integration of diverse, heterogeneous data sources and reasoning paradigms to support intelligent decision making. Their work has focused on heterogeneous databases and mediation technology, agent technology, multimedia databases and systems, and inferential reasoning (data mining, uncertainty inference, and planning).

Leveraging work from a previous project, the heterogeneous ontology management environment (HOME), UMD developed a stand-alone heterogeneous ontology object server (HOS) that functions as an ontology object cache for simplifying ontology application usage. Rather than parsing and instantiating an ontology graph during initialization, HOS client applications save precious time by retrieving cached ontology graph objects from the HOS. HOS supports a Web Ontology Language (OWL) export mode for external OWL-view centric ontology browsers and editors (such as Swoop and Protégé) and can run as either a stand-alone server or as an embedded Java service page application which includes a web-based client user interface.

UMD developed a cross-platform prototype set of client/server interface modules. In effect, a mediator, the AWE data proxy, serves to integrate BED's object weather effects information as transparent area-of-interest (AOI) map overlays on CISD's decision support tool interface.

The AWE data proxy functions as a query translation and execution layer that links CISD's intelligence officer's decision support tool interface, BED's IWEDA application database and weather forecasts data, and UMD's HOS. In short, the AWE data proxy provides a fast layer which allows seamless embedding of IWEDA generated mission component weather effect impact displays (map overlays) directly into the intelligence officer's decision support tool interface. Figure 9 shows the general architecture of the integrated system.

The terms "IWEDA MySQL database" will be used hereafter to denote the MySQL rules database and the forecast meteorological database required to execute IWEDA that were provided with the demo IWEDA software application.

### **4.1 AWE Overview**

The AWE data proxy provides a concise text based TCP/IP socket interface for simple network client usage (such as with the decision support interface's Visual Basic software layer). AWE renders result sets as tokenized answer lists where the entries are typically the data point text or file universal resource locator (URL) references. Graphical interfaces, such as CISD's decision support interface tool, can automate the interaction by rendering AWE query constraints

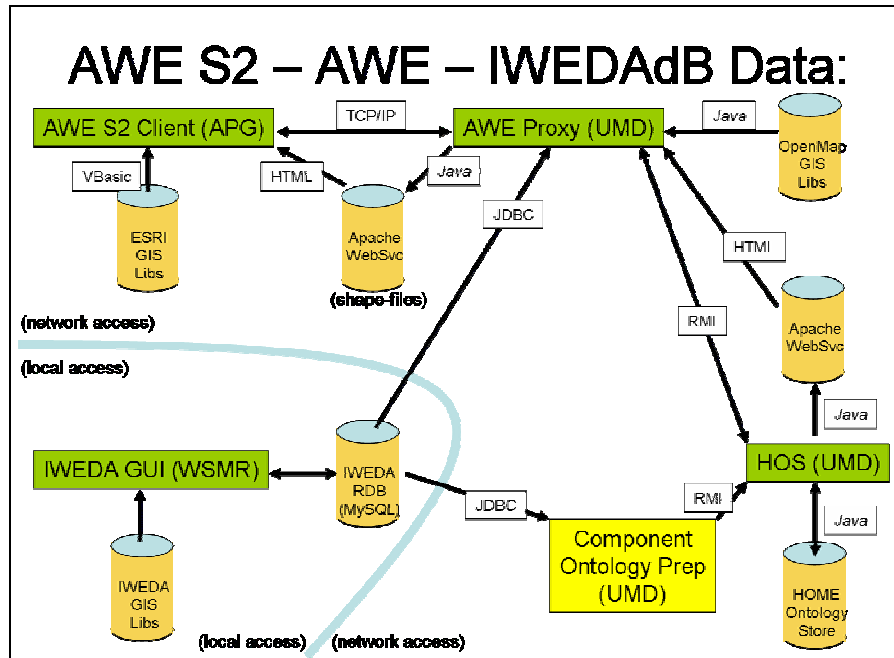


Figure 9. Architecture of integrated system.

(mission and time-slice references) as control object entries (e.g., a series of buttons, list-box, and combo-box entries) for simple point and click usage. As the user specifies and AWE answers a weather impact query, the decision support interface software updates the user's AOI map display to include the newly generated weather-impact shape files as transparent overlays.

Behind the scenes, AWE maintains a hybrid weather effect impact geographic information system (GIS) data structure (a complex quad tree referenced hash-table and tree-set object collection) derived from IWEDA's MySQL database. During initialization, the decision support tool interface renders mission and time-slice query constraint controls from AWE's response to some base queries (`getMissionList` and `getTimeStampList`, respectively). As for the mission list, these entries represent the root graph nodes from an ontology extracted from the IWEDA database, embellished, and stored in the HOS for simple client access. At query execution time, AWE extracts the underlying mission entry object "sub-nodes" from the IWEDA ontology to generate a rule reference query, the answer to which is then used as a filter over the GIS data structure. At this point, if the user prompted for map overlay(s) (`getWeatherOverlay`), then the AWE proxy publishes a series of shape files (encoding the underlying filtered GIS rule impact data) and reports the data file URL sent back to the decision support tool interface client. The decision support tool interface, in turn, reads and displays the target shape files as embedded transparent AOI map overlays. If the user prompted for a grid-cell detail page (`getWeatherDetailPage`), then AWE publishes a web page listing the target grid cell's mission filtered weather effect impact rule details (components impacted for specific grid cell locations).

In short, AWE generates weather effects impact shape binary files with grid detail information as a result of user interface inputs from the decision support tool and provides this data to the decision support tool via web pages. As the user picks one or more mission filters via the decision support tool interface controls, the decision support tool interface prompts AWE to render shape files, which are then viewed as embedded transparent AOI map overlays.

## **4.2 IWEDA Database Table Descriptions and Usage**

The provided demo IWEDA database consists of 22 tables (comp\_names, comp\_rules, def\_comp\_rules, def\_rules, def\_subsys\_rules, def\_system\_rules, domain, fcasts, free\_rule\_nums, impacts, map\_extents, param, pt\_fcasts, pt\_impacts, pt\_map\_extents, rules, subsys\_comps, subsys\_names, subsys\_rules, system\_names, system\_rules, systems\_subsys) which wrap the target interest area's calculated weather effects and weapons system component decomposition for a given time span.

Typically, during system preparation, IWEDA administrators query the weather database for a target location grid (the map extents table) and projected time span (the fcasts table). The administrators then invoke a process which evaluates weather parameter threshold rules (the rules and param tables) for each time slice per grid cell in the target location grid (wind speed exceeds X, temperature below Y, etc.). The threshold rules that are exceeded are then referenced by grid cell per time slice in the impacts table.

This clever use of exceeded threshold encoding provides a compact representation that can be easily reconstructed to reveal the offending weather parameter(s) for any defined grid cell per time-slice combination.

At run time, the IWEDA GUI user invokes various controls to define the weather effects view constraints from the underlying IWEDA database tables. These constraints are then applied to the grid cell per time-slice rule reference queries in the impacts table, the results of which, drive the red-amber-green overlay production.

The use of dedicated, relational database driven, component hierarchy tables appears to limit the current IWEDA GUI to a common, local data source, local maintenance paradigm. Separating the component hierarchy source(s) from the generated threshold grid could allow a more useful, distributed application of remote clients. While clients may be sharing a common threshold grid, they might be monitoring the end effects on radically different asset sets. For example, two commanders covering the same interest area (one concerned with area defense, the other with medical evacuations) could share the same threshold grid while considering extremely different asset effect pictures.

## **4.3 IWEDA Ontology**

UMD developed the initial AWE demo IWEDA ontology by first extracting a graph of all the IWEDA database rule entry names (linked to the various system, subsystem, and component

roots [figure 10]) through a custom HOME extension module. The extension module embellished the graph with additional hierarchy links (edges [figure 11]) and then stored the instantiated ontology, via HOS, for use with the AWE data proxy server demo.

```
SELECT DISTINCT name FROM system_rules;
SELECT DISTINCT name FROM subsys_rules;
SELECT DISTINCT name FROM comp_rules;
```

Figure 10. Three base IWEDA graph node extraction queries.

```
SELECT DISTINCT a.name, b.name AS subSysName
FROM systems_subsys a, subsys_names b
WHERE ABS( a.sub_id) = b.id;

SELECT DISTINCT a.name, b.name as CompName
FROM subsys_comps a, comp_names b
WHERE ABS( a.sub_id) = b.id
```

Figure 11. Two base IWEDA graph edge extraction queries.

The IWEDA ontology used by the AWE data proxy bypasses the expense and reliance on the original IWEDA encoded mission/component hierarchy data. Rather, the AWE data proxy client administrator extracts the base graph once, embellishes it accordingly, and injects one or more variations into the HOS for use in a variety of end-user mission applications. Given a base set of mission resources (vehicles, weapons, people, etc.), a user exploring the effects on a battle maneuver may use the resources differently than one proposing a humanitarian evacuation operation. Hence, the same base IWEDA graph can be extended with mission-specific embellishments, preloaded into the HOS, and used according to the end user's run-time preferences. This effectively separates the dependence on the original IWEDA hierarchy encoding, allowing for a simplified data model with reduced entry redundancy.

#### 4.4 AWE Proxy

Where the existing IWEDA application relies on system-subsystem component hierarchies encoded in the MySQL database, the AWE data proxy uses an ontology graph extracted from the IWEDA rules database, embellished with additional hierarchy links (edges), and stored in the HOS for convenience. During initialization, the AWE data proxy retrieves the target IWEDA ontology from the HOS.

Since IWEDA's weather effect impact information is, in effect, static in the current implementation (e.g., it is preloaded through a selection and initialization process), the AWE data proxy renders a GIS-based data cache of the IWEDA's weather effects data during the first



startup. This is then serialized to a binary object which allows for quicker initialization during subsequent reboots.

As users request weather overlays, the AWE data proxy first generates the relevant impact rule filter set through one additional IWEDA database query (constrained via elements extracted from the mission ontology). Using the ontology saves at least five IWEDA database queries per user request. The AWE data proxy then applies the impact rule filter set and the time-slice constraints to the GIS data cache (basically, a set intersection operation) updates the grid colors on an internal OpenMap display layer, prompts the OpenMap Environmental Systems Research Institute module to write shape files for the display layer objects to the target web server (or file reference), and finally returns the shape file URL(s) to the client interface. Figure 12 is a rendering of the final shape file sent to the client interface.

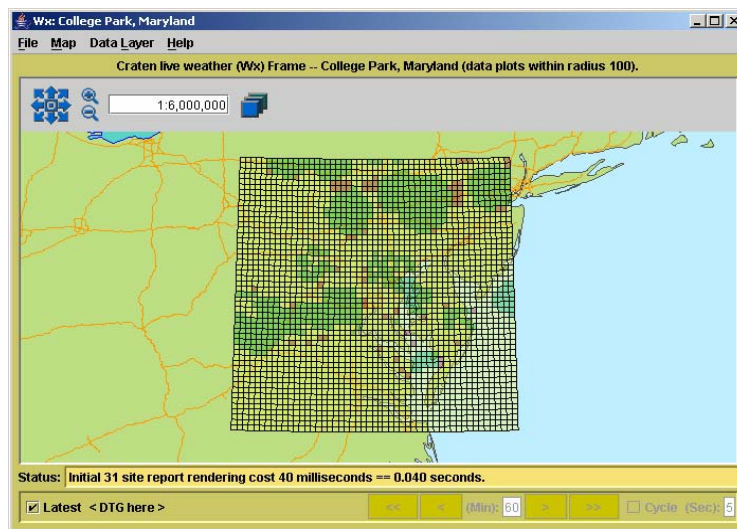


Figure 12. AWE rendered weather effect overlay grid.

When users request a weather detail page, the AWE data proxy builds the target input rule filter set (as before). It then performs a set intersection with the target grid cell data, renders a web page listing the weather effect details, and returns the URL to the client interface (figure 13).

#### 4.5 Decision Support Tool Interface With AWE Proxy

The AWE enhanced decision support tool interface provides the user with an embedded map overlay display of BED's weather impact data. This was not previously possible; users had to manually cross reference separate application displays, a costly procedure with both time and accuracy concerns.

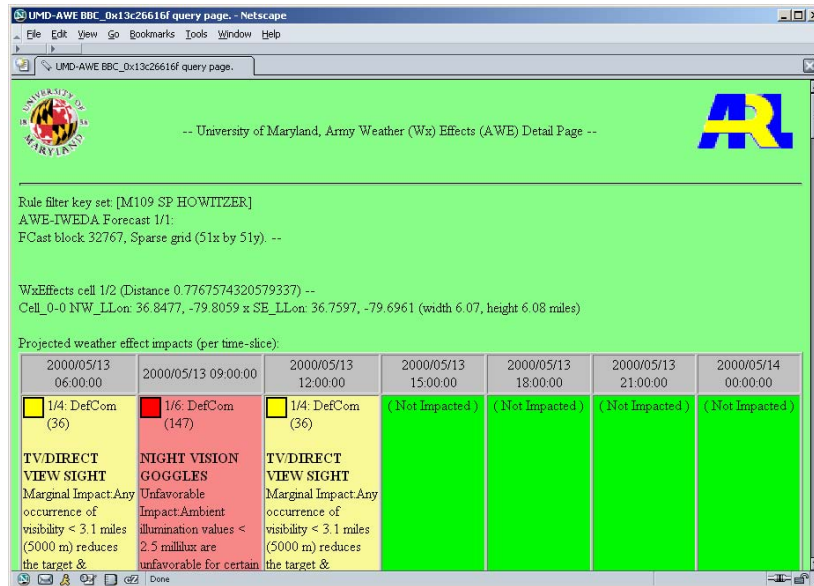


Figure 13. AWE rendered grid cell detail page.

The decision support tool interface is actually two separate applications—one for the UDOP and one for the Intel visualization display (which includes the search and asset screen). When the decision support tool interface is started, three screens similar to figure 14 will appear.

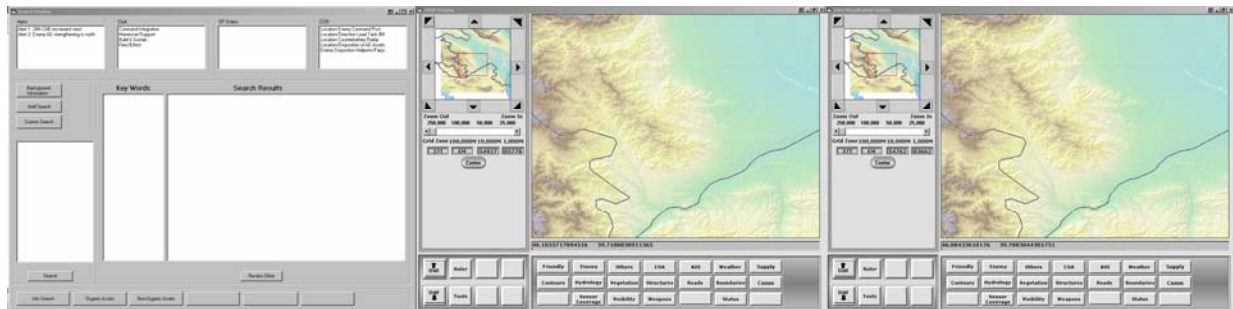


Figure 14. Three-screen decision support tool interface display.

During the initialization process, the decision support tool interface will try to make a connection to the AWE proxy on the host given in the host option and communicating over the port in the port option, with the user name and password provided in the user name and password options. When a connection is successfully established with AWE, the decision support tool interface sends a request for a set of mission lists and all the time stamps currently available for interrogation. (See section 4.1 for a description of how AWE produces the mission lists and time stamps.)

Overlays are turned on/off by the user via the overlay buttons located directly beneath the large map, as seen in figure 15. This provides the flexibility to tailor the display to meet the user's current needs. To choose a weather overlay, the user must select the weather overlay button. Weather overlays are mainly used as quick snapshots for planning purposes and will most likely not be left visible for a long period of time. So, the user is given the option to turn on the last generated weather overlay or generate a brand new weather overlay. When generating a new weather overlay, a weather controls box appears to the left of the large map (figure 15).

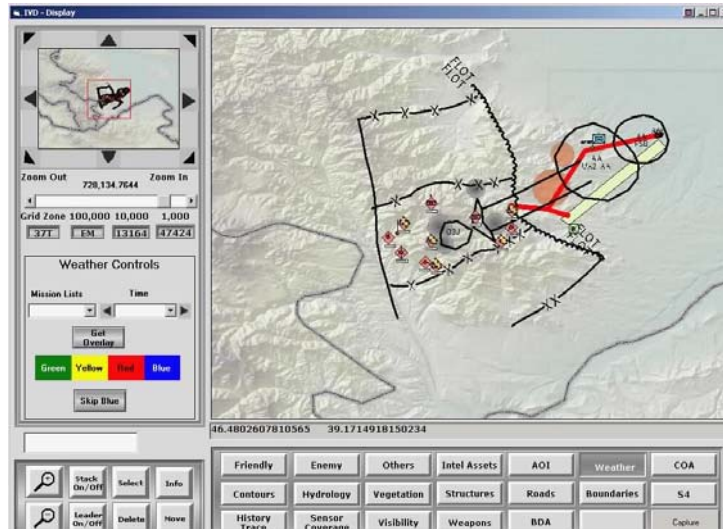


Figure 15. Weather controls box.

The user chooses a mission from the missions pull-down list (figure 16) and a time slice from the time pull-down list (figure 17).

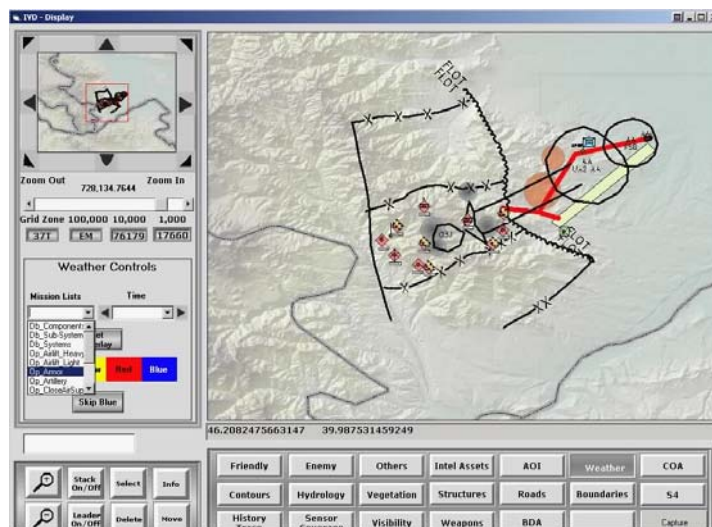


Figure 16. Mission lists.

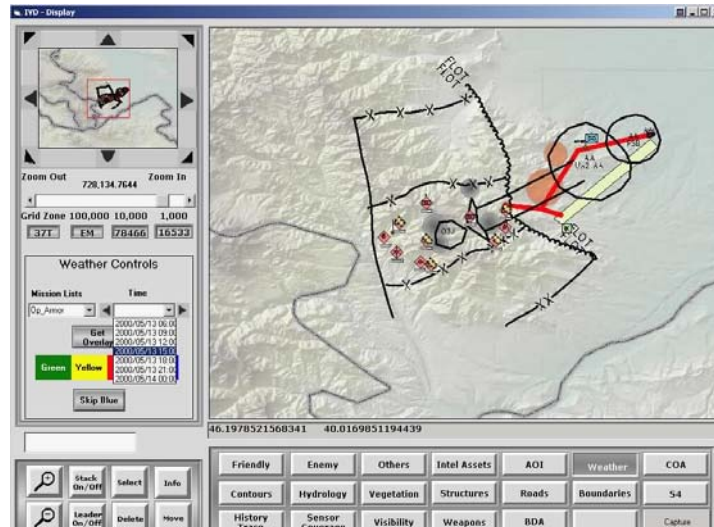


Figure 17. Time stamps.

The user then selects the “Get Overlay” to retrieve the overlay from the AWE. The “Get Overlay” button function calls the AWE application program interface (API) method “getWeatherOverlay.” As described in section 4.1, the AWE produces a set of geospatial shape files containing the red-amber-green conditions for the chosen mission at the chosen time slice and publishes them to a URL that is reported back to the decision support tool interface. The decision support tool interface downloads the files from the URL and displays them as semitransparent layers on the AOI map (figure 18).

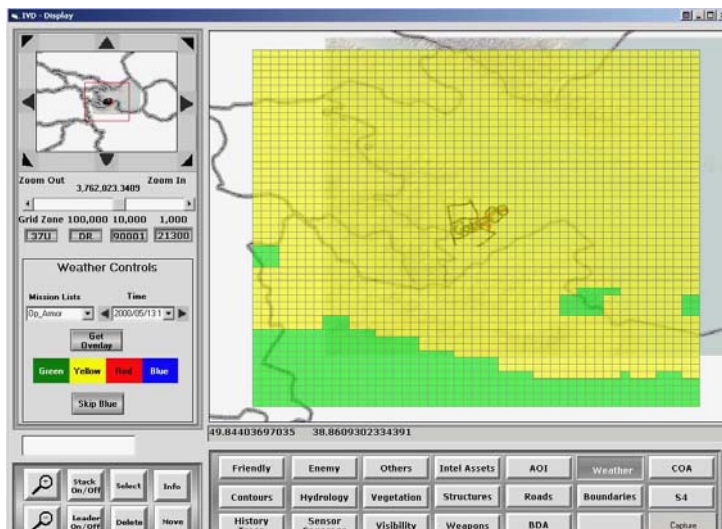


Figure 18. Weather overlay.

---

## 5. Future Directions

---

While this weather effects overlay generation method was developed specifically for the IWEDA data, it could easily be adapted to display many different data view map overlays. Hence, this work can serve as an inexpensive, platform independent, nonproprietary solution template for future battlefield data view integration efforts. To further the applicability and usability of this integration effort, web service description layer wrappers could be written for the underlying AWE data proxy server and the HOS. This would render the AWE system available for a wider array of automated query systems.

Since the beginning of this project, the IWEDA database has undergone a complete redesign. It would be desirable to incorporate these changes and automate the underlying IWEDA database mission data preload step. The current (manual) data preload steps can be time consuming and error prone. Automating the weather database data extraction, modification, and IWEDA database load procedures could render this as a relatively simple and timely GUI initialization step. Another time-saving step would be adding an IWEDA update trigger that would easily automate the cache-block regeneration process on-the-fly. Currently, this process is being done by the AWE data proxy during initial start-up and remains static until the server is rebooted.

Many enhancements are being considered for the decision support tool interface, not the least of which is porting the software to Java. This will provide greater flexibility and portability to the interface, allowing for a smoother integration with other battlefield tools. As a Java application, the installation and execution of the interface will become much simpler, with less dependence on third-party software. An immediate and relatively simple enhancement to the tool would be to provide the user with grid-cell weather effect impact details via a Web page pop-up view port. The information and the vehicle (API call) are currently available through the AWE interface; it is just a matter of capturing the user's request through a right button click on one of the displayed weather impact grid cells to retrieve and display this information.

NO. OF  
COPIES ORGANIZATION

1 DEFENSE TECHNICAL  
(PDF INFORMATION CTR  
ONLY) DTIC OCA  
8725 JOHN J KINGMAN RD  
STE 0944  
FORT BELVOIR VA 22060-6218

1 US ARMY RSRCH DEV &  
ENGRG CMD  
SYSTEMS OF SYSTEMS  
INTEGRATION  
AMSRD SS T  
6000 6TH ST STE 100  
FORT BELVOIR VA 22060-5608

1 INST FOR ADVNCD TCHNLGY  
THE UNIV OF TEXAS  
AT AUSTIN  
3925 W BRAKER LN  
AUSTIN TX 78759-5316

1 DIRECTOR  
US ARMY RESEARCH LAB  
IMNE ALC IMS  
2800 POWDER MILL RD  
ADELPHI MD 20783-1197

3 DIRECTOR  
US ARMY RESEARCH LAB  
AMSRD ARL CI OK TL  
2800 POWDER MILL RD  
ADELPHI MD 20783-1197

ABERDEEN PROVING GROUND

1 DIR USARL  
AMSRD ARL CI OK TP (BLDG 4600)

NO. OF  
COPIES ORGANIZATION

- 2 DIR USARL  
AMSRD ARL CI EE  
D HOOK  
J RABY  
WSMR NM 88002
  
- 1 UNIV OF MD  
T ROGERS  
FACULTY RESEARCH ASSIST  
3247 A V WILLIAMS BLDG  
COLLEGE PARK MD 20742
  
- 1 UNIV OF MD  
V S SUBRAHMANIAN  
DIRECTOR  
UMIACS & PROFESSOR CS  
3143 A V WILLIAMS BLDG  
COLLEGE PARK MD 20742

INTENTIONALLY LEFT BLANK.