ARMY RESEARCH LABORATORY

Utilizing JavaSpaces for Distributing Weather Research and Forecasting Model Output and Meteorological Assimilation Data Ingest System Data and Their Use in the Local Analysis and Prediction System

by Stephen Kirby

ARL-TR-3963 **November 2006**

Approved for public release; distribution is unlimited.

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
White Sands Missile Range, NM 88002-5501

ARL-TR-3963 **November 2006**

Utilizing JavaSpaces for Distributing Weather Research and Forecasting Model Output and Meteorological Assimilation Data Ingest System Data and Their Use in the Local Analysis and Prediction System

Stephen Kirby
Computational and Information Sciences Directorate

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
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.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*
November 2006

2. REPORT TYPE
Draft

3. DATES COVERED (From - To)

4. TITLE AND SUBTITLE
Utilizing JavaSpaces for Distributing Weather Research and Forecasting Model Output and Meteorological Assimilation Data Ingest System Data and Their Use in the Local Analysis and Prediction System

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Stephen Kirby

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
Computational and Information Sciences Directorate
Battlefield Environment Division (ATTN: AMSRD-ARL-CI-EM)
White Sands Missile Range, NM 88002-5501

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-3963

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1197

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
ARL-TR-3963

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The Future Force can benefit from a means for inhomogeneous computer systems to access data objects through the network. A platform that supports this paradigm is JavaSpaces, because they are based on the Java Virtual Machine; the concept "write once, run anywhere" applies. This report illustrates how a Weather Research and Forecasting (WRF) model output file can be split into objects based on the dimension of the variable and then placed in a JavaSpace, where the data objects are accessible to clients and can be reassembled into their original WRF output file format. A methodology, whereby WRF data is used as the "background" model for the Local Analysis and Prediction System (LAPS), is discussed; the converse, of utilizing LAPS output to initialize a fine-scale WRF model run is also presented.

15. SUBJECT TERMS
JavaSpaces, Weather Research and Forecasting Model, WRF, Local Analysis and Prediction System, LAPS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stephen Kirby
U	U	U	SAR	32	19b. TELEPHONE NUMBER (*Include area code*) 505-678-7474

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures **iv**

Summary **1**

1. Introduction **3**

2. JavaSpaces and WRF Variable Objects **3**

3. Initializing LAPS with WRF **10**

4. LAPS Execution **10**

5. Executing WRF Utilizing LAPS as Background **19**

6. Conclusions **19**

7. Future Considerations **19**

References **21**

Acronyms **22**

Distribution List **24**

List of Figures

Figure 1. A schematic illustrating the placement of WRF output objects into a JavaSpace and clients accessing these objects. 6

Figure 2. A screen capture showing the server (upper left hand window) placing a WRF variable entry into a JavaSpace and the client Java agent (lower right hand window) finding a 1-D entry and processing it. 9

Figure 3. N-S cross section of wind speed valid time (VT) 30 May 2006, 0300 UT. 12

Figure 4. N-S cross section of wind speed VT 30 May 2006, 0600 UT. 12

Figure 5. N-S cross section of wind speed VT 30 May 2006, 0900 UT. 13

Figure 6. N-S cross section of wind speed VT 30 May 2006, 1200 UT. 13

Figure 7. N-S cross section of wind speed VT 30 May 2006, 1500 UT. 14

Figure 8. N-S cross section of wind speed VT 30 May 2006, 1800 UT. 14

Figure 9. N-S cross section of wind VT 30 May 2006, 0900 UT. 15

Figure 10. A sounding within the LAPS domain. 16

Figure 11. Total precipitable water (in cm). 17

Figure 12. The 850 hPa temperatures (°C) for the 30 May 2006, 0900 UT LAPS case. 18

Summary

The Future Force will face challenges in terms of making data accessible to clients in a seamless manner. Clients, running on computers likely featuring differing operating systems (OS), will need to access inhomogeneous datasets on servers with different OSs. An intermediate storage area accessible to these widely varying clients is required. Such a storage area can be constructed utilizing JavaSpaces, an interface developed by Sun Microsystems, Inc. Because JavaSpaces are built based on the Java Virtual Machine (VM), any client who can run the Java VM will be able to access data objects within a JavaSpace. Similarly, any server that can run the Java VM will be able to place objects into a JavaSpace.

With the JavaSpaces interface, applications may be developed that allow the user to do the following: 1) create templates to represent data "objects," 2) build objects based on these templates, 3) place these objects in a JavaSpace, 4) take these objects from a JavaSpace, and if desired, 5) notify a client when a certain type of object has been inserted into the JavaSpace.

One Future Force application, the Nowcast, is a weather forecast that is to be updated frequently with new observations and large-scale model background data. The first step in this process is to obtain large-scale Global Forecast System (GFS) model initialization data to initialize a relatively coarser scale (6-km horizontal resolution) Weather Research and Forecasting (WRF) model run. This "outer" WRF run data will, in turn, be used to initialize a Local Analysis and Prediction System (LAPS) model run.

At a remote computer, a client wishing to run LAPS will require this "outer" WRF model output data for initialization. Obtaining an entire WRF output file for this purpose would require significant bandwidth. Also, if the WRF output file were in a database, the client would require the database server to be up, as well as need to know the protocols for database access. To circumvent these hindrances, this report shows how a WRF output file can be split into objects, each representing one WRF output variable. Each object will contain the variable name, its dimensions, and the data block tied to that variable.

One of the methods specified in the JavaSpaces interface is "notify." With this method, Java agents can continuously poll a JavaSpace searching for a particular object. For this application, three Java agents can run on the client that search for one-dimensional, two-dimensional, or three-dimensional WRF object variables in the JavaSpace. When such an object is found, the agent performs a take on the space and proceeds to extract the data portion from the entry and write it to a file on the client, still in its original serialized format. The agent then converts the serialized format to a readable text format. Another agent on the client monitors for the presence of all the files needed to regenerate the WRF output file into its original network Common Data format (netCDF). When all of the required files are found, the agent runs the program that recreates the WRF output netCDF file. The client now has a file (WRF netCDF) that is suitable for input to a model such as LAPS. Once the LAPS model has been executed, the client can generate plots using the LAPS-supplied graphics program, LAPSPLOT. Sample LAPSPLOT output graphics are provided showing both cross sections and planar views of various LAPS output parameters.

This report also briefly covers how the LAPS model output can be used as input to a finer scale (2-km horizontal resolution) WRF model run. This cascade of a coarse scale WRF run used to initialize LAPS and then LAPS used to initialize a finer scale WRF run, performed cyclically, comprises one Nowcast scenario.

For this report, Internet data sources, such as the Meteorological Data Ingest System (MADIS), were employed. Such Internet sources may not be available to the Future Force. Thus, alternate data sources, such as the Air Force Weather Agency, may need to be examined as sources for entities such as large-scale model background data and satellite data.

1. Introduction

The net-centric Future Force will require a means for heterogeneous (in terms of operating system (OS)) clients to asynchronously query heterogeneous server(s) for a wide variety of datasets, including meteorological. Fault tolerance must be inherent, i.e., the loss of a server should not obstruct a client from obtaining the requisite data. The JavaSpace paradigm addresses these issues: a JavaSpace can be described as "a shared, network-accessible repository for objects" and a JavaSpace application as "a collection of processes cooperating via the flow of objects into and out of one or more *spaces* (*1*)." One Future Force application is a "Nowcast," which will be comprised of short (1–3 h), frequently updated forecasts based on the Weather Research and Forecasting (WRF) model. WRF is first run at a coarser scale, comprising a domain approximately 8 km on a side. The output from this model run serves as initialization data for the Local Analysis and Prediction System (LAPS) model, which also assimilates a multitude of datasets including surface, upper air, and satellite. The boundary conditions from the LAPS model run then serve to initialize the finer-scale WRF Nowcast run, which will be nested inside both the coarser-scale WRF domain and the LAPS domain. A demonstration of how WRF model output can be converted into objects and made accessible to a generic WRF client through a JavaSpace and then reconstituted into a format amenable for use as a background field for LAPS will be presented in this report, along with methods for generating objects out of Meteorological Assimilation Data Ingest System (MADIS) data and passing them through a JavaSpace. A look at how LAPS can be used as a means to initialize and set up WRF runs on a continuing basis, thus yielding a Nowcast, is also covered.

2. JavaSpaces and WRF Variable Objects

JavaSpaces can be utilized in various ways. Two ways that apply here are as a means for either shared distributed communication or object storage. JavaSpaces are elegantly simple as evidenced by the diminutiveness of its specification (*2*):

```
public interface JavaSpace{
      Lease write (Entry entry, Transaction txn, long lease)
            throws TransactionException, RemoteException;

             long NO_WAIT=0;

       Entry read (Entry tmpl, Transaction txn, long timeout)
            Throws UnusableEntryException,TransactionException,
            InterruptedException,RemoteException;

       Entry take (Entry tmpl, Transaction txn, long timeout)
            Throws UnusableEntryException,
            TransactionException,InterruptedException,
            RemoteException;
```

```
                    Entry takeIfExists (Entry tmpl, Transaction txn, long
                    timeout)
                          Throws UnusableEntryException,
                          TransactionException,InterruptedException,
                          RemoteException;

                    EventRegistration notify (Entry tmpl, Transaction txn,
                    RemoteEventListener listener, long lease, MarshalledObject
                    handback)
                          Throws TransactionException, RemoteException;

                    Entry snapshot (Entry e) throws RemoteException;
              }
```

As seen in the aforementioned specification, there are four main methods: read, write, take, and notify. For this effort, entry classes for one-dimensional (1-D), two-dimensional (2-D), and three-dimensional (3-D) variables extracted from WRF output files have been designed. For example, the 2-D entry class is as follows:

```
public class WrfEntry_2d implements Entry{

 public String variable;
 public byte[] arry;
 public String central_latitude;
 public String central_longitude;
 public String time_stamp;
 public Integer Time;
 public Integer s_n;
 public Integer w_e;

 public WrfEntry_2d(){
 }

public WrfEntry_2d(String name, byte[] arr, String lat,String lon,String
ts,Integer dim1,Integer dim2,Integer dim3){
   variable=name;
   arry=arr;
   central_latitude=lat;
   central_longitude=lon;
   time_stamp=ts;
   Time=dim1;
   s_n=dim2;
   w_e=dim3;
 }
```

Each field of an entry must be declared public. Here the variable name; data array; grid center latitude and longitude; number of forecast periods; time the WRF output was generated; and *x*- and *y*-grid dimensions have all been defined. Note also the required empty constructor as well as a constructor to set each of the fields.

JavaSpaces are designed to overcome some of the deficiencies in distributed communication between computers. For example, a Remote Procedure Call (RPC) requires the following (*3*):

1. The client to know who it is calling
2. The client to know where the server is
3. The client and server to be active at the same time
4. The client and server to agree in detail on what form the call will take
5. The client and server to agree in detail on how the call will be made

JavaSpaces surmount these potential bottlenecks by virtue of "loose coupling," that is, "processes interact indirectly through a space and not directly with other processes (*2*)." Thus, for example, a server can write an object to a JavaSpace with some Lease time (i.e., how long it is to remain in the space). A client can then search this space by virtue of a take (which removes the entry) or a read (which obtains a copy of the entry), either of which requires a (entry) template as an argument. This template can be very specific by defining all of the fields or quite generic by defining only one of the fields and wildcarding (set to null) all others. The use of a generic template greatly increases the chance of matching an entry in the space.

In this application, the 1-, 2-, and 3-D variables within the WRF network Common Data Format (netCDF) file are extracted using netCDF file operators, the data serialized, and the objects then written to the JavaSpace (see figure 1.). The generic server process, which writes the objects to the space, takes arguments consisting of the relevant variable, its dimensions, the number of forecast periods, and integers defining each dimension size. Three client agents monitor the JavaSpace, using the JavaSpace notify method, searching for 1-, 2-, or 3-D WRF output variable entries. Each agent blocks until it finds its object type, at which point it performs a take on that object with the agent, and then deserializes the data and writes it to a file. Another agent blocks until all of the requisite data files have been garnered by the client. Upon discovery of the required files, it launches a program to reconstruct the separate 1-, 2-, and 3-D variable data files back to a single netCDF format file. Finally, when an adequate number of netCDF files are present, LAPS can be executed.

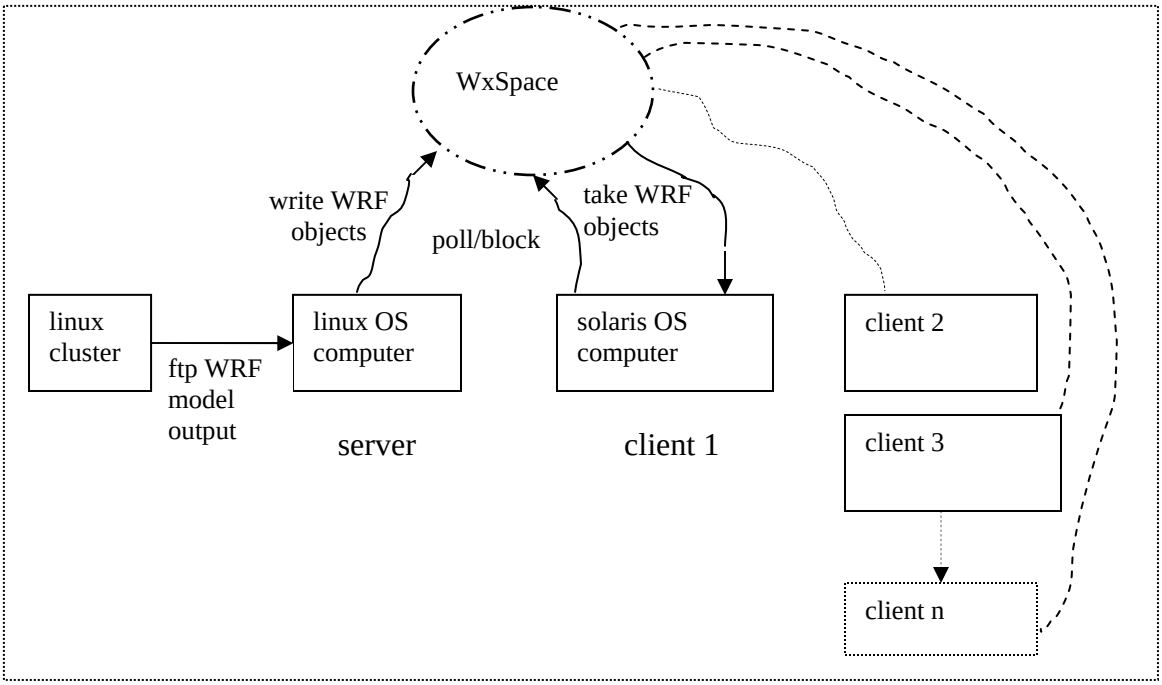

Figure 1. A schematic illustrating the placement of WRF output objects into a JavaSpace and clients accessing these objects.

The WRF objects written to the JavaSpace include the following:

1. 1-D objects
   - model run time
   - upper weight for vertical stretching
   - lower weight for vertical stretching
   - inverse d(eta) values between full (w) levels
   - inverse d(eta) values between half (mass) levels
   - d(eta) values between full (w) levels
   - d(eta) values between half (mass) levels
   - eta values on half (mass) levels
   - eta values on full (w) levels
   - two extrapolation constants
   - leapfrog time filter constant
   - inverse *x* grid length
   - inverse *y* grid length
   - time weight constant for small steps
   - zeta at model top
   - three 2nd order extrapolation constants

- model time step
- depths of centers of soil layers
- thicknesses of soil layers
- pressure top of the model

2. 2-D objects
    - land use category
    - perturbation dry air mass in column
    - base state dry air mass in column
    - nest position
    - water vapor mixing ratio at 2 m
    - temperature at 2 m
    - potential temp at 2 m
    - surface pressure
    - *u*-component at 10 m
    - *v*-component at 10 m
    - land mask—1 for land; 0 for water
    - sea ice flag
    - surface runoff (in mm)
    - underground runoff
    - dominant vegetation category
    - dominant soil category
    - vegetation fraction
    - ground heat flux
    - snow water equivalent
    - physical snow depth
    - canopy water
    - sea surface temperature
    - map scale factor on mass grid
    - map scale factor on *u*-grid
    - map scale factor on *v*-grid
    - Coriolis sine latitude term
    - Coriolis cosine latitude term
    - local sine of map rotation

- local cosine of map rotation
- terrain height
- surface skin temperature
- accumulated grid scale accumulation
- accumulated cumulus scale accumulation
- downward shortwave flux at ground surface
- downward longwave flux at ground surface
- latitude, south is negative
- longitude, west is negative
- albedo
- soil temperature
- landmask—1 for land, 2 for water
- reciprocal of Monin-Obhukov length
- planetary boundary layer (PBL) height
- upward heat flux at the surface
- upward moisture flux at surface
- latent heat flux at surface
- flag indicating snow coverage—1 for snow coverage

3. 3-D objects
    - U—*x* wind component
    - V—*y* wind component
    - W—*z* wind component
    - perturbation geopotential
    - base state geopotential
    - perturbation potential temperature
    - perturbation pressure
    - base state pressure
    - water vapor mixing ratio
    - cloud water mixing ratio
    - rain water mixing ratio
    - ice mixing ratio
    - snow mixing ratio
    - graupel mixing ratio

- soil temperature
- soil moisture
- soil liquid water

Figure 2 is a screen capture of activity involving WRF objects and a JavaSpace, in which the window at the upper left displays activity on the server and the window at the lower right displays activity on the client. In this instance, the server is processing each of the 1-D data blocks by serializing each variable entry, placing it in the JavaSpace, and then blocking to give the client discovery and processing time. On the client side, a Java agent, "notify_1d," continuously scans the JavaSpace; as soon as a 1-D entry is found, a take is performed and the data is written in its current serialized form to the client machine. The agent then converts the serialized (binary) file into a text file in preparation for the reconstitution into a WRF netCDF file later.

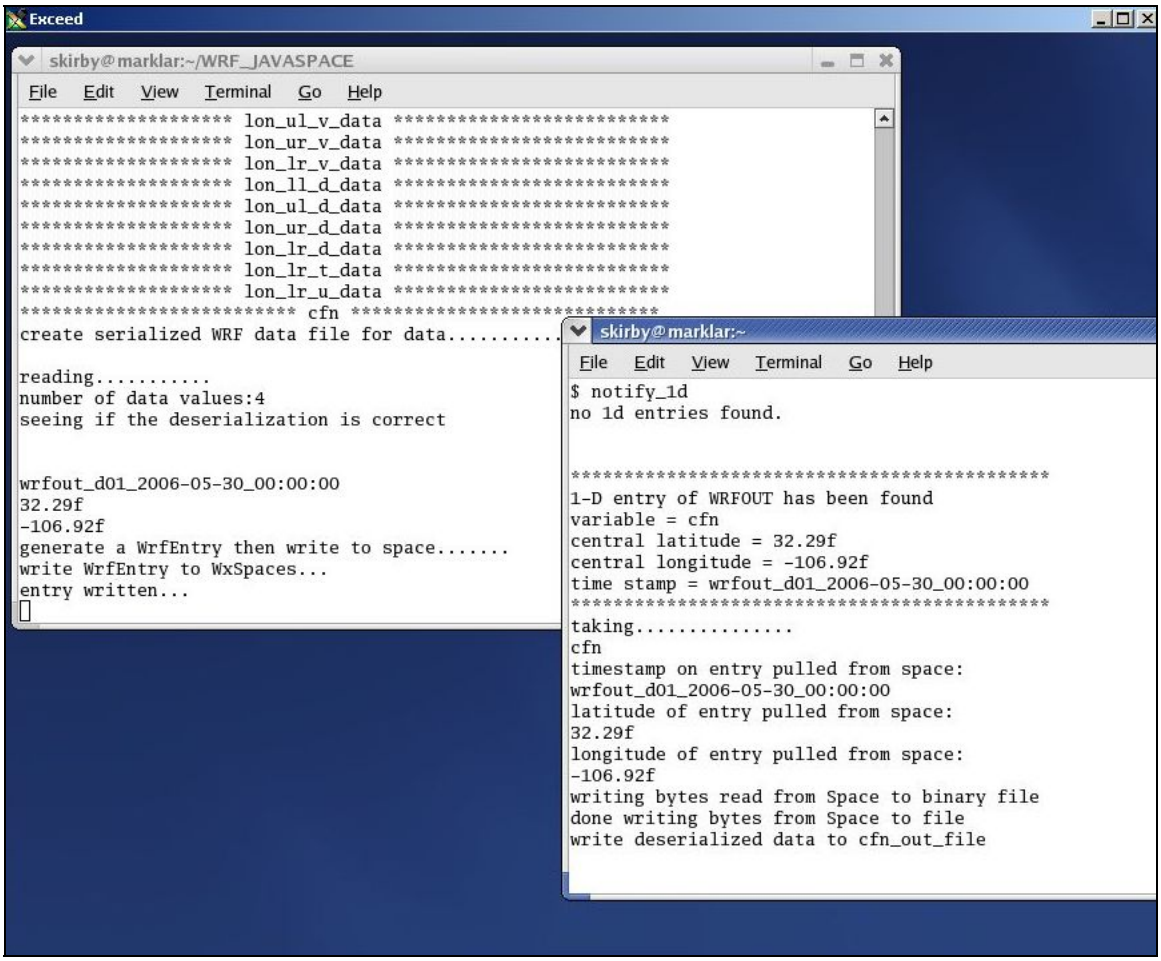

Figure 2. A screen capture showing the server (upper left hand window) placing a WRF variable entry into a JavaSpace and the client Java agent (lower right hand window) finding a 1-D entry and processing it.

The process to place MADIS data into the JavaSpace is quite similar to that described for WRF. For this portion, the netCDF operator, ncdump, is also utilized and generates text files of data from MADIS mesonet netCDF data for latitude, longitude, report time, 2-m temperature, dew

point, relative humidity, wind direction, wind speed, visibility, and elevation data. Because each of these file sizes are relatively small, all of the data types for one time period can be combined into one serialized data file prior to placement into the JavaSpace.

3. Initializing LAPS with WRF

LAPS requires a coarser scale background model for initialization. Among those that have been used in LAPS are the Rapid Update Cycle (RUC) (*4*), GFS (*5*), Navy Operational Global Atmospheric Prediction System (NOGAPS) (*6*), Mesoscale Model Version 5 (MM5) (*7*), Regional Atmospheric Modeling System (RAMS) (*8*), and WRF (*9*) models. The WRF model was chosen as the model to initialize LAPS, because it is considered the next generation of the MM5 and also because of the author's prior experience executing WRF in parallel on a local Linux cluster.

The large-scale model background utilized for this effort was the Global Forecast System (GFS) model. This "outer" WRF domain contained 144 by 144 grid points with 6-km grid spacing and 31 levels in the vertical. For the 0000 universal time (UT) model run, the analysis fields and the 3-h, 6-h, 9-h, 12-h, 15-h, 18-h, 21-h, and 24-h forecasts were obtained from the National Center for Environmental Prediction (NCEP). In order to mesh with the 3-h LAPS cycle time employed, WRF was run for seven 3-h forecast periods: 0000–0300 UT, 0300–0600 UT, 0600–0900 UT, 0900–1200 UT, 1200–1500 UT, 1500–1800 UT, and 1800–2100 UT on a 16-node dual processor Linux cluster.

Due to the Linux cluster being located on a private network, it was not possible to communicate from it to the JavaSpace, thus the WRF output was transferred to a Linux computer, which acted as a server (see figure 1). In fact, the server and client were deliberately chosen to be systems with differing OSs to further illustrate the concept of loose coupling. JavaSpaces are built on top of the Java Virtual Machine; thus, the concept of "write once, run anywhere" applies.

4. LAPS Execution

The essence of LAPS is its ability to assimilate a very large number of different data types in order to provide the most realistic depiction of an atmospheric volume over short (1–3 h) time periods. A non-exhaustive list of data that LAPS can ingest includes radio acoustic sounding system (RASS), high density winds (HDW) (derived from Geostationary Operational Environmental Satellite (GOES) data), buoy, Meteorological Aerodrome Report (METAR) (surface reports), satellite sounder, satellite imagery, radiosonde observation (RAOB), aircraft automated reports (ACAR), pilot reports (PIREP), Next Generation Weather Radar (NEXRAD), global positioning system (GPS) (from which moisture profiles can be inferred), profiler, radiometer data, as well as data from the background model mentioned in section 2. For this application, MADIS (managed by National Oceanic and Atmospheric Administration (NOAA) Global Systems Division (GSD)) provided ACAR, HDW, mesonet, METAR, polar orbiting satellite (POES), profiler, radiometer, RAOB, and satellite radiance data. To insure that local effects were accounted for, a utility was built to convert White Sands Missile Range mesonet

“Surface Automated Measurement System” (SAMS) data into an intermediate format, known as the LAPS surface observation (LSO) format. The ability to convert to a text intermediate format obviated the need to convert the data into a netCDF format.

The LAPS domain was nested inside the WRF domain, with 400 by 400 grid points at 2-km grid spacing and 21 vertical levels. This domain was also centered at the Las Cruces, NM, Airport. The fields produced by LAPS were output at 0300 UT, 0600 UT, 0900 UT, 1200 UT, 1500 UT, and 1800 UT.

LAPS output can be visualized with a National Center for Atmospheric Research (NCAR) graphics-based utility, LAPSPLOT. Some sample plots from a LAPS run for 30 May 2006 are shown in figures 3–12. Figures 3–8 are cross sections, oriented north (N) to south (S) across the domain displaying isotachs. Note the development of a low-level jet by 0600 UT and its dissipation by 1800 UT.

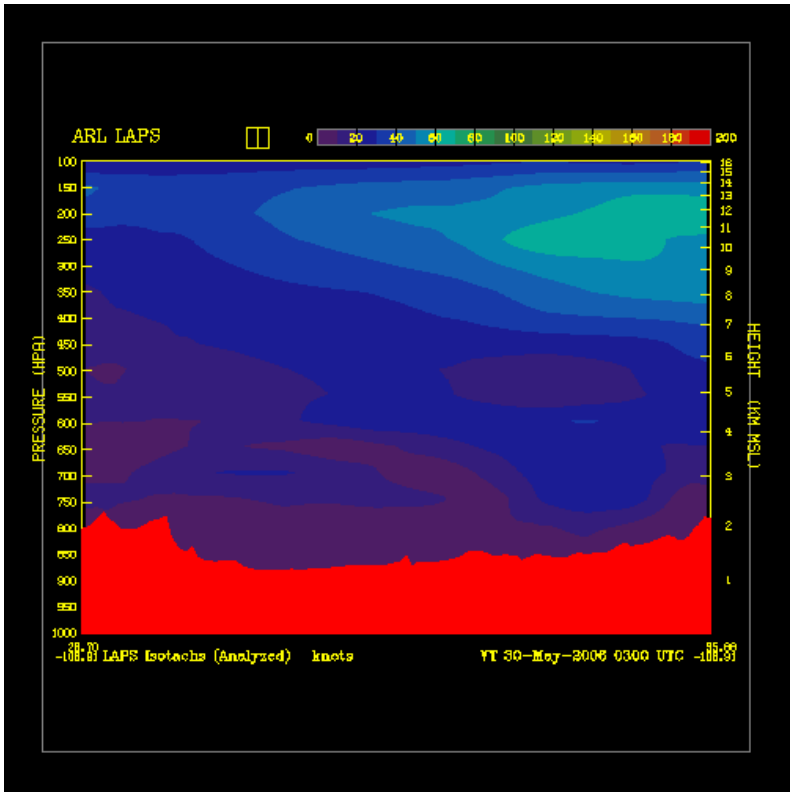

Figure 3. N-S cross section of wind speed valid time (VT) 30 May 2006, 0300 UT.

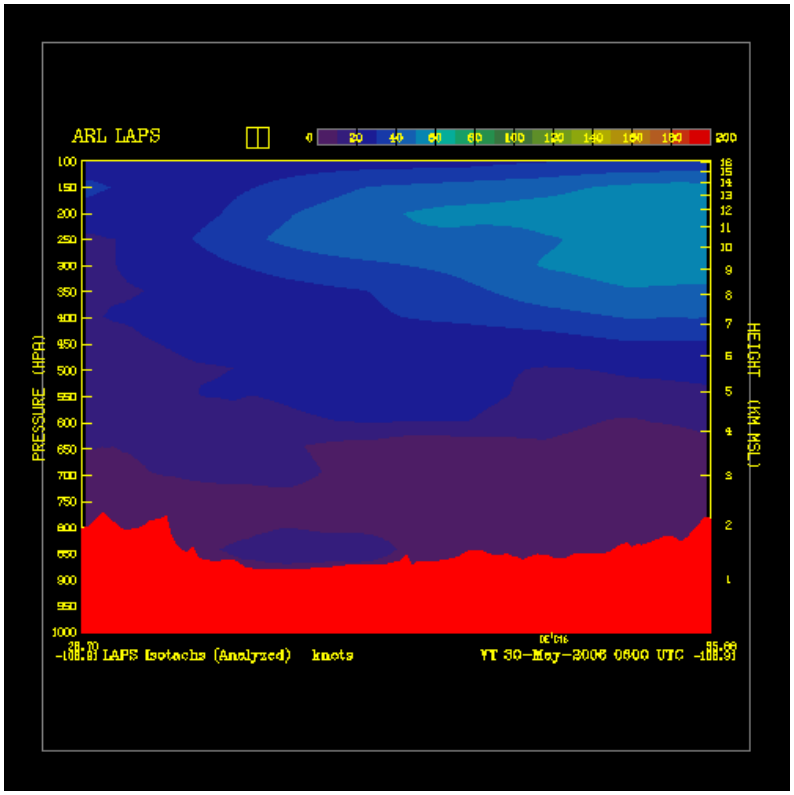

Figure 4. N-S cross section of wind speed VT 30 May 2006, 0600 UT.

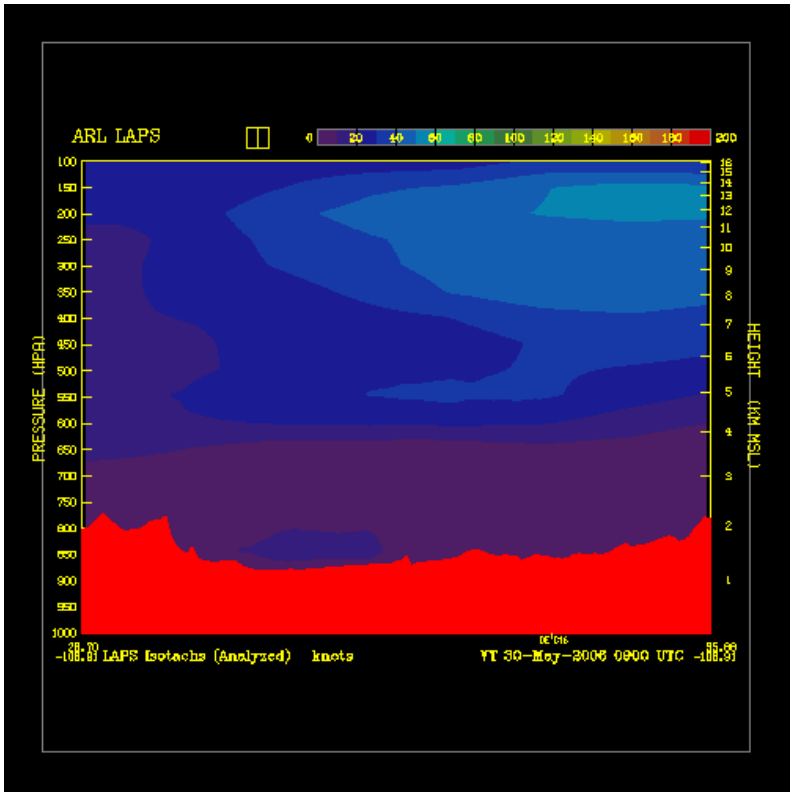

Figure 5. N-S cross section of wind speed VT 30 May 2006, 0900 UT.

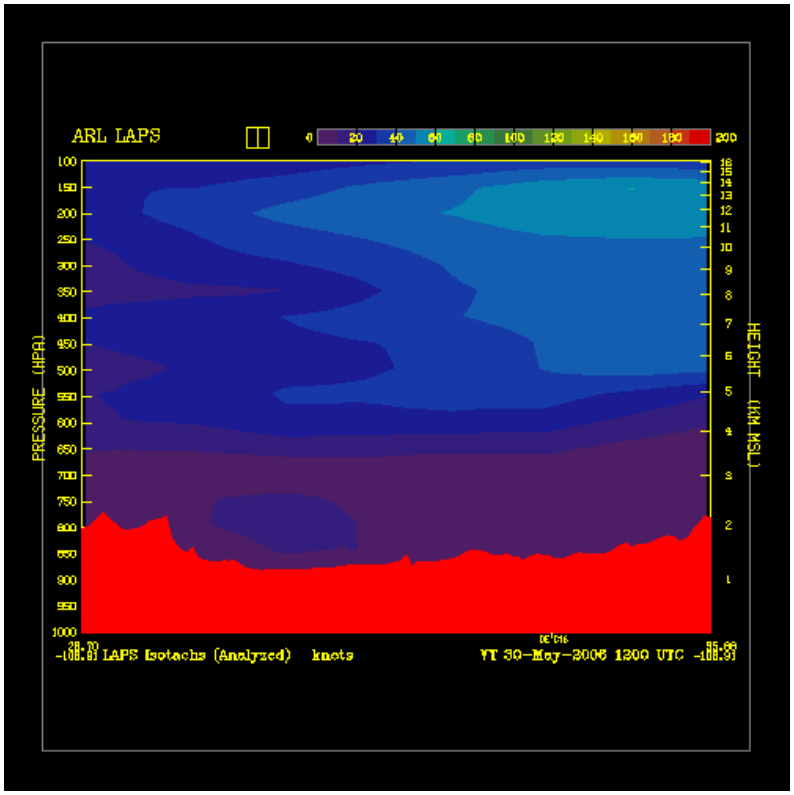

Figure 6. N-S cross section of wind speed VT 30 May 2006, 1200 UT.

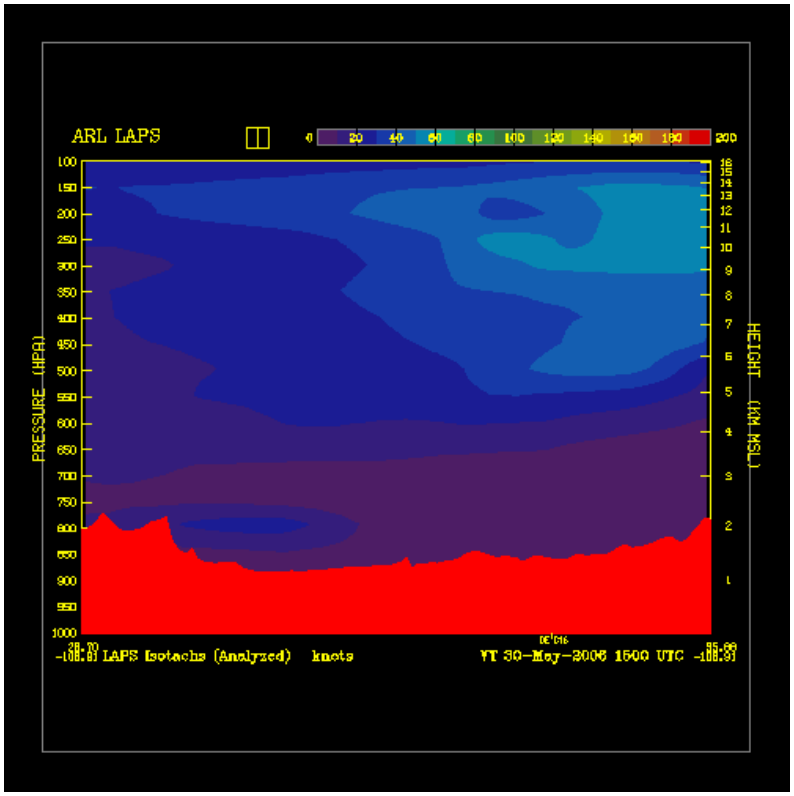

Figure 7. N-S cross section of wind speed VT 30 May 2006, 1500 UT.

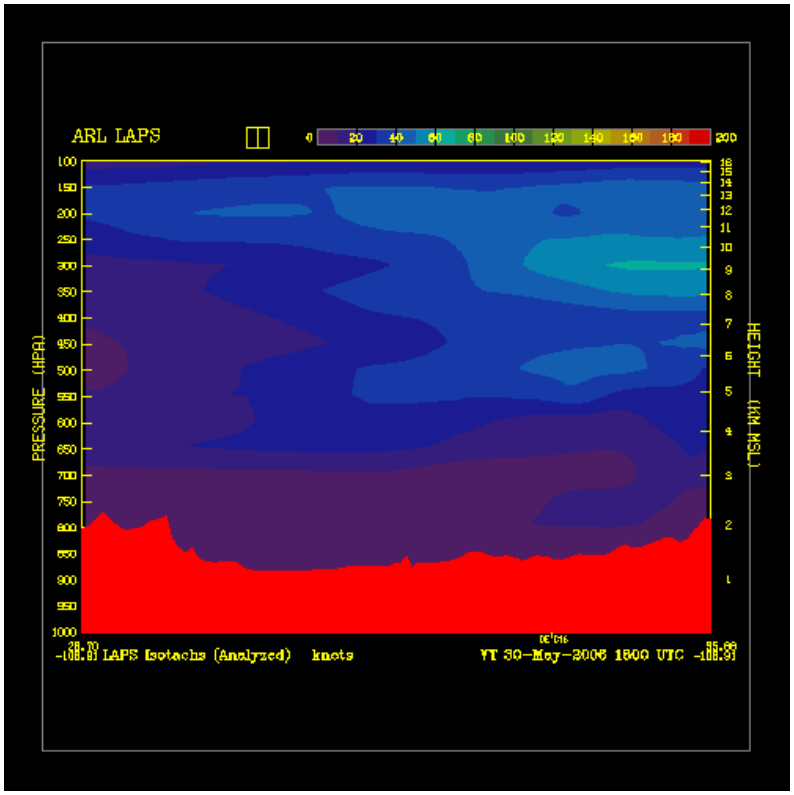

Figure 8. N-S cross section of wind speed VT 30 May 2006, 1800 UT.

Figure 9 is a N-S oriented cross section of winds. Note the veering of the winds between 450 hPa and 700 hPa at the southern end of the domain.

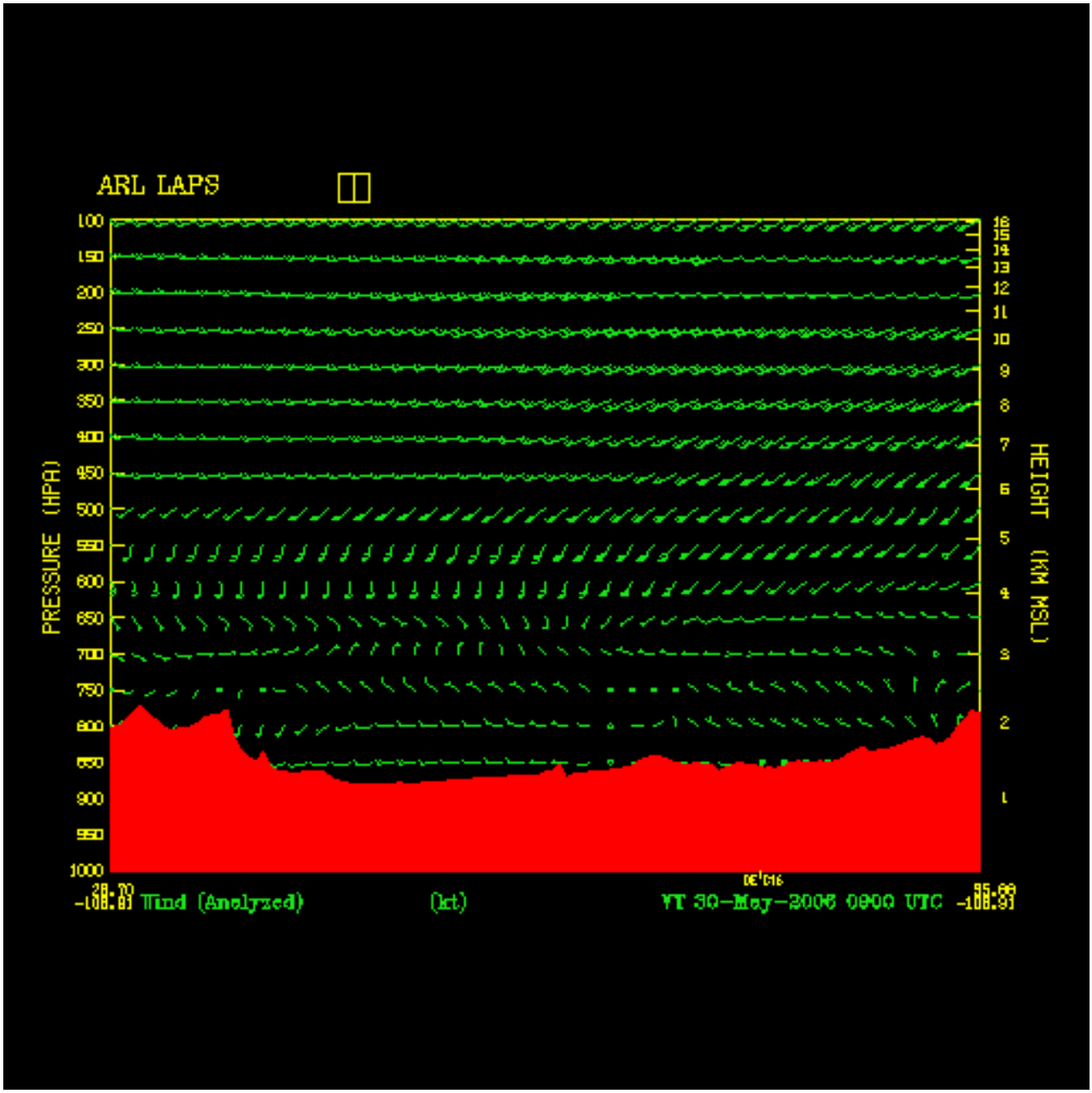

Figure 9. N-S cross section of wind VT 30 May 2006, 0900 UT.

LAPSPLOT can also produce sounding diagrams, as shown in figure 10.

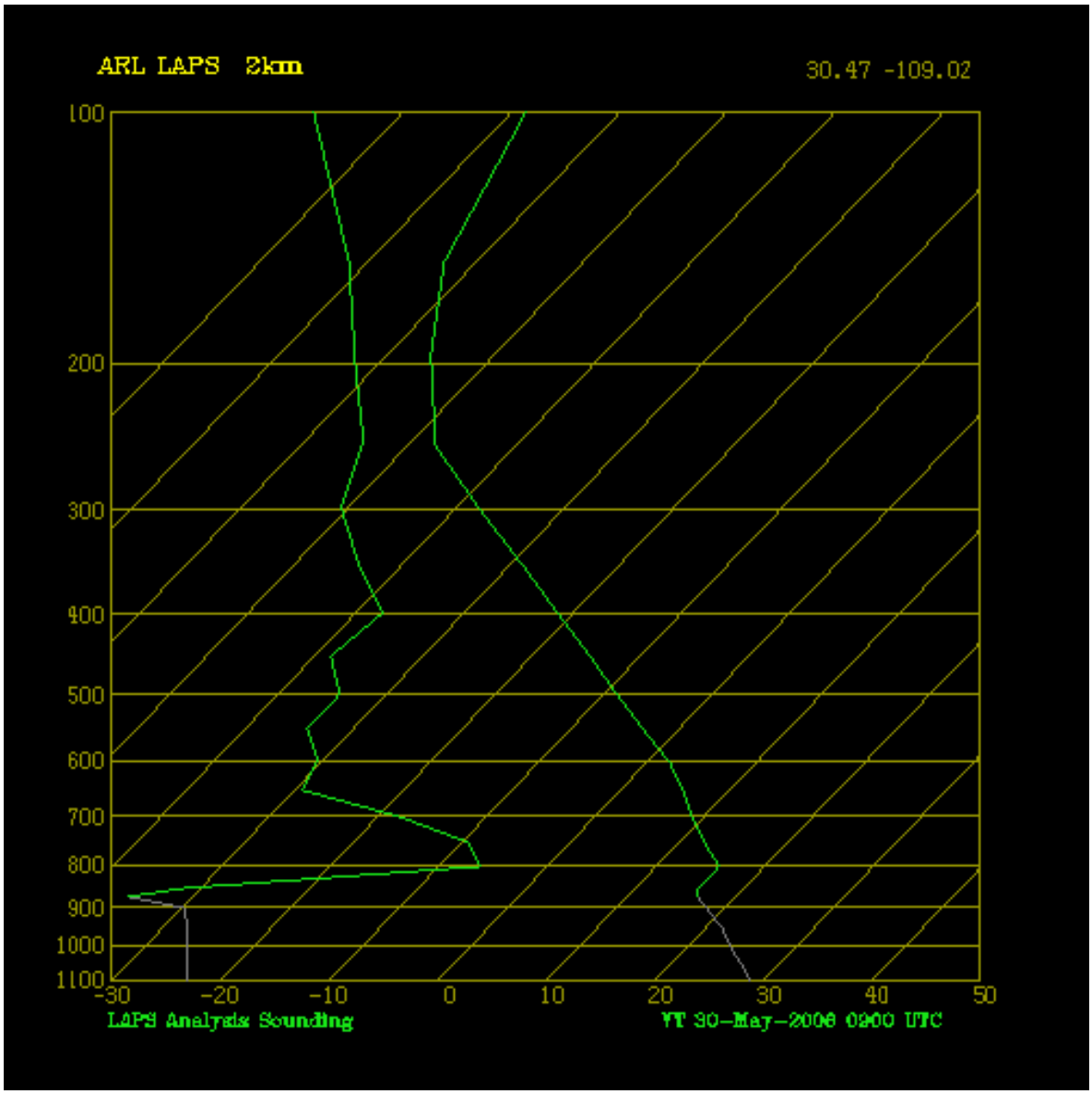

Figure 10. A sounding within the LAPS domain.

NOTE: The *x*-axis represents temperature (°C) and the *y*-axis represents pressure (hPa).

Figure 11 illustrates LAPS output showing the integrated total precipitable water over the entire domain. This LAPS run preceded the monsoonal flow typical to the region in July and August; thus, the deepest moisture is restricted to the extreme eastern portions of the domain.

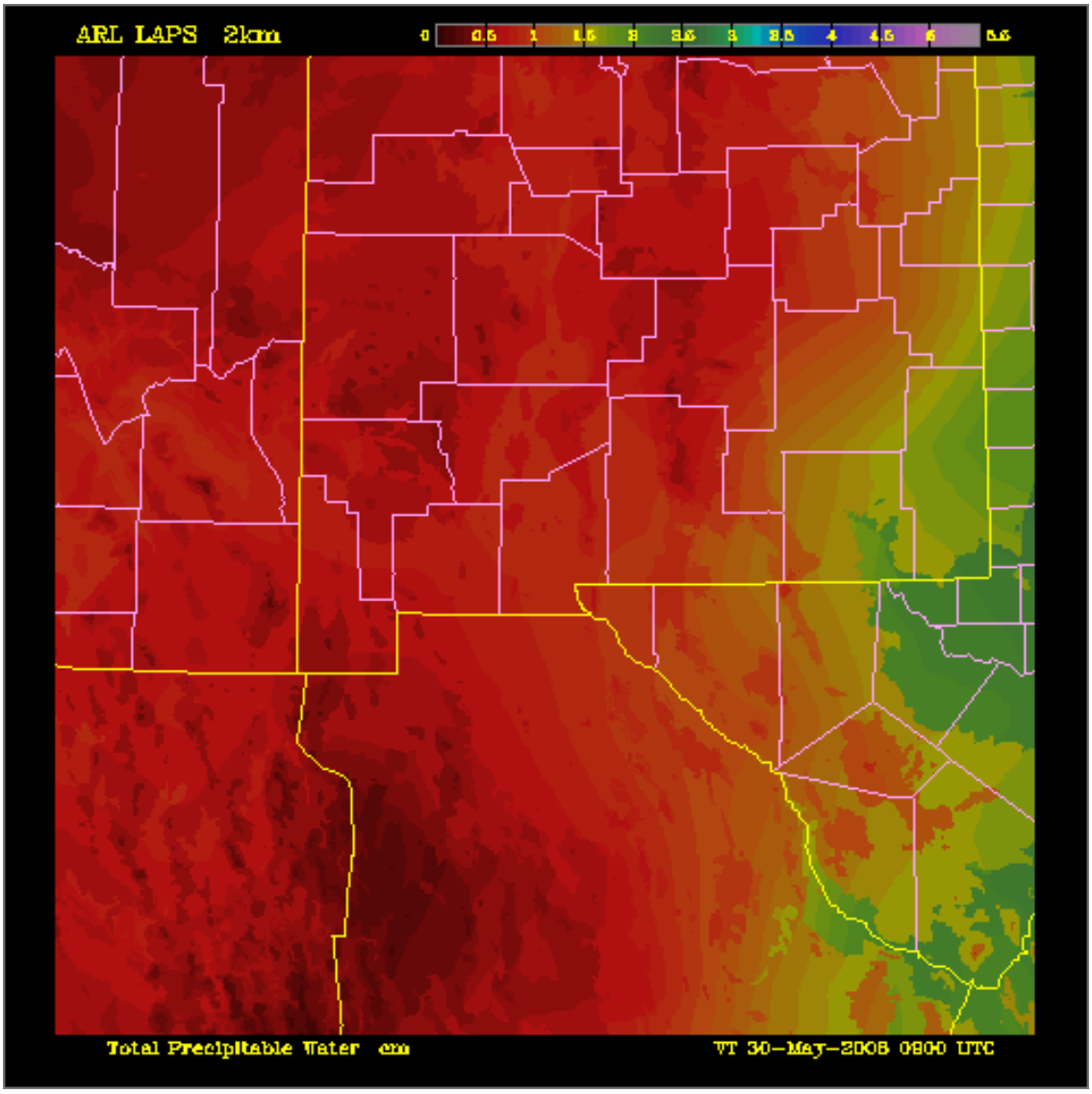

Figure 11. Total precipitable water (in cm).

Figure 12 depicts the 850 hPa (~1500 m level) temperatures for the 30 May 2006, 0900 UT LAPS case.

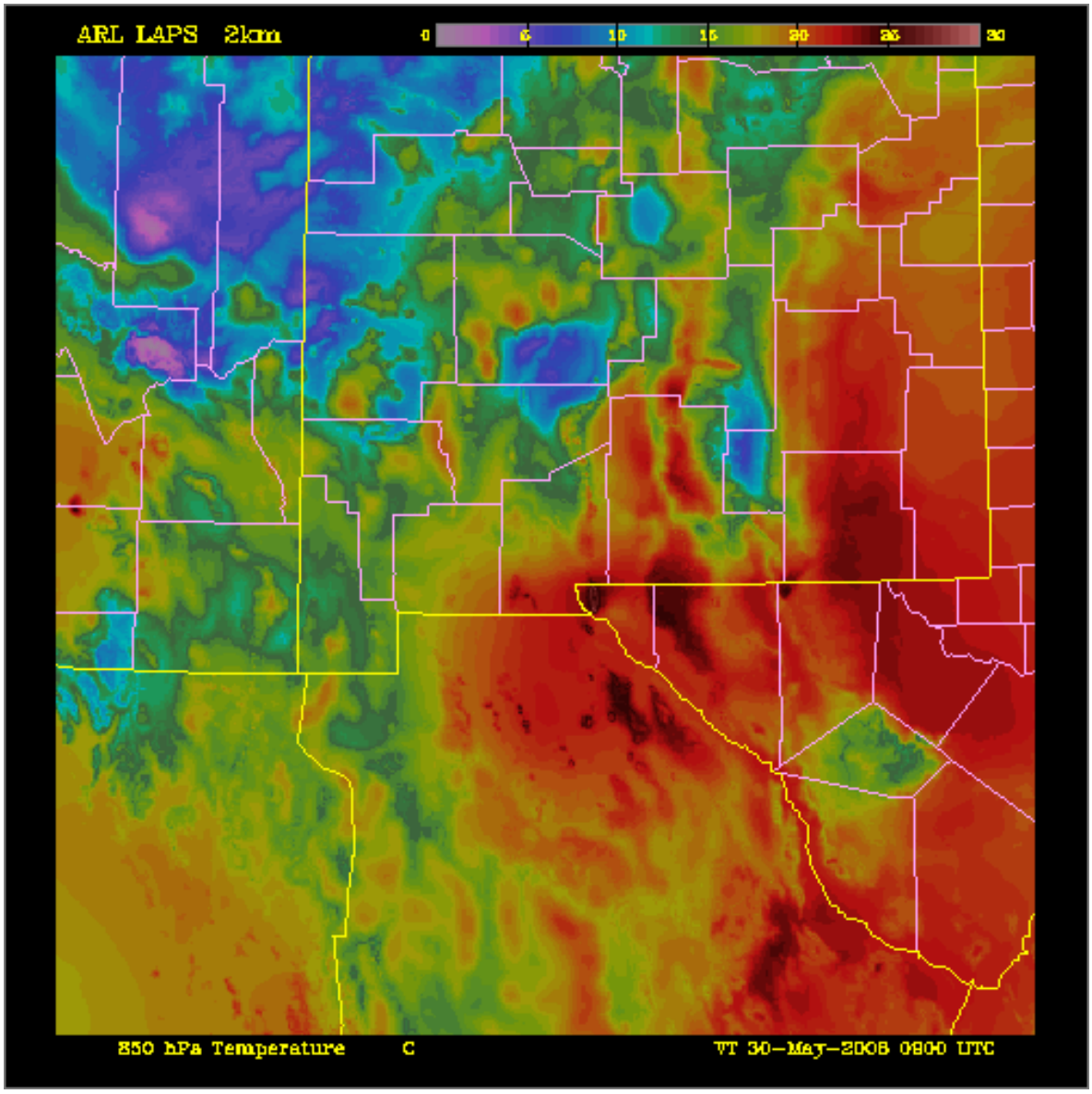

Figure 12. The 850 hPa temperatures (°C) for the 30 May 2006, 0900 UT LAPS case.

5. Executing WRF Utilizing LAPS as Background

Next, the resultant LAPS output files served as large-scale background initialization data for yet another WRF run, whose domain was set to be inside the LAPS domain. Specifically, the size was 350 by 350 grid points and 2-km grid spacing with 31 vertical levels. North American Model (NAM) model data was also introduced to provide three additional soil temperature layers required for the NCEP, Oregon State University, Air Force, Hydrologic Research Lab (NOAH) Land Surface Model (LSM), one component of WRF, to successfully execute. Again 3-h forecast windows were used and Advanced Research WRF (ARW) output times were 0300–0600 UT, 0600–0900 UT, 0900–1200 UT, 1200–1500 UT, and 1500–1800 UT.

6. Conclusions

As a result of this study, an application has been developed whereby the following are possible:

1. A WRF output netCDF file can be split into separate files, each representing one WRF variable that has been serialized and then placed into a JavaSpace.
2. Agents running on a client machine can monitor the JavaSpace for 1-, 2-, or 3-D variables. When found, the variable can be taken from the JavaSpace, written first to a file in its original (binary) format, and then converted to a text format.
3. The text files can be reassembled into a netCDF file suitable for input to a model such as LAPS.

Although the finer-scale WRF run will provide the greatest fidelity in terms of horizontal resolution, it was instructive to visualize the LAPS data, which will serve as input to the WRF, by means of LAPSPLOT, with some of its capabilities illustrated in section 3.

7. Future Considerations

For this effort MADIS data from NOAA GSD supplied all of the upper air and satellite observations and the majority of the surface observations. For the Future Force, this Internet source will likely not be available; thus, alternate sources for data will need to be pursued. For example, the Air Force Weather Agency can supply the WRF large-scale initialization files needed for input to LAPS; however, their Gridded Binary (GRIB) format will require conversion to the netCDF format first. Candidates for satellite data include the SeaSpace TeraScan hardware/software system or the NOAAport data feed available from Unidata (*10*). Although not tested in this effort, radar data can be processed by LAPS and should be examined; however, its applicability in a Future Force scenario is unclear.

Also, a Future Force Nowcast application is anticipated to have an outer domain model with 80 by 80 grid points at a 5-km grid resolution with an inner nest of 100 by 100 points at a 1-km resolution. Execution time for a model such as WRF, run at such a fine horizontal scale, needs to be investigated to determine what size cluster will be required for “reasonable” run times.

References

1. Freeman, E.; Hupfer, S.; Arnold, K. *JavaSpaces Principles, Patterns, and Practice*; Addison-Wesley Longman Ltd.: Essex, UK, 1999.
2. Halter, S.L. *JavaSpaces Example by Example*; Prentice Hall PTR: Upper Saddle River, NJ, 2002.
3. McClain, J. (staff engineer) *Introduction to JavaSpaces Technology*. Sun Microsystems, Inc., PowerPoint presentation (http://www.javasig.com/Archive/lectures/JavaSIG-Jini-IntroToJavaSpaces.pdf).
4. Rapid Update Cycle (RUC) model. http://maps.fsl.noaa.gov/ (accessed 5 Oct 2006).
5. Global Forecast Model (GFS). http://www.emc.ncep.noaa.gov/modelinfo/index.html (accessed 5 Oct 2006).
6. Navy Operational Global Atmospheric Prediction System (NOGAPS). http://www.nrlmry.navy.mil/sawmill/index.html (accessed 5 Oct 2006).
7. Mesoscale Model Version 5 (MM5). http://www.mmm.ucar.edu/mm5/ (accessed 5 Oct 2006).
8. Regional Atmospheric Modeling System (RAMS). http://www.srh.noaa.gov/ssd/nwpmodel/html/rams.htm (accessed 5 Oct 2006).
9. Weather Research and Forecasting (WRF). http://www.wrf-model.org/index.php (accessed 5 Oct 2006).
10. Dumais, R., research meteorologist, U.S. Army Research Laboratory, Computational and Information Sciences Directorate, Battlefield Environment Division: White Sands Missile Range, NM. Personal communication, September 2006.

Acronyms

1-D	one-dimensional
2-D	two-dimensional
3-D	three-dimensional
ACAR	aircraft automated reports
ARW	Advanced Research WRF
GFS	Global Forecast System
GOES	Geostationary Operational Environmental Satellite
GPS	global positioning system
GRIB	Gridded Binary
GSD	Global Systems Division
HDW	high density winds
LAPS	Local Analysis and Prediction System
LSM	Land Surface Model
LSO	LAPS surface observation
MADIS	Meteorological Assimilation Data Ingest System
METAR	Meteorological Aerodrome Report
MM5	Mesoscale Model Version 5
N	north
NAM	North American Model
NCAR	National Center for Atmospheric Research
NCEP	National Center for Environmental Prediction
netCDF	Network Common Data Format
NEXRAD	Next Generation Weather Radar
NOAA	National Oceanic and Atmospheric Administration
NOAH	NCEP, Oregon State University, Air Force, Hydrologic Research Lab
NOGAPS	Navy Operational Global Atmospheric Prediction System
OS	operating system
PBL	planetary boundary layer
PIREP	pilot reports

POES	polar orbiting satellite
RAMS	Regional Atmospheric Modeling System
RAOB	radiosonde observation
RASS	radio acoustic sounding system
RPC	Remote Procedure Call
RUC	Rapid Update Cycle
S	south
SAMS	Surface Automated Measurement System
UT	universal time
VM	Virtual Machine
VT	valid time
WRF	Weather Research and Forecasting

Distribution List

	Copies
DARPA ATTN IXO S WELBY 3701 N FAIRFAX DR ARLINGTON VA 22203-1714	1
MIL ASST FOR ENV SCI OFC OF THE UNDERSEC OF DEFNS FOR RSRCH & ENGRG R&AT E LS PENTAGON RM 3D129 WASHINGTON DC 20301-3080	1
OFC OF THE SECY OF DEFNS ATTN ODDRE (R&AT) THE PENTAGON WASHINGTON DC 20301-3080	1
SCI & TECHNLGY CORP 10 BASIL SAWYER DR HAMPTON VA 23666-1340	1
NATL CTR FOR ATMOS RSRCH ATTN NCAR LIBRARY SERIALS PO BOX 3000 BOULDER CO 80307-3000	1
US ARMY TRADOC BATTLE LAB INTEGRATION & TECHL DIRCTRT ATTN ATCD B 10 WHISTLER LANE FT MONROE VA 23651-5850	1
TECOM ATTN CSTE DTC CL APG MD 21005-5057	1
COMMANDING GENERAL US ARMY AVN & MIS CMND ATTN AMSTM RD W C MCCORKLE REDSTONE ARSENAL AL 35898-5000	1
US ARMY CRREL ATTN CRREL GP R DETSCH ATTN CECRL GP M MORAN ATTN CERCL SI E L ANDREAS 72 LYME RD HANOVER NH 03755-1290	3

	Copies
US ARMY DUGWAY PROVING GROUND DPG METEOROLOGY DIV ATTN J BOWERS WEST DESERT TEST CENTER DUGWAY UT 84022-5000	1
US ARMY INFO SYS ENGRG CMND ATTN AMSEL IE TD F JENIA FT HUACHUCA AZ 85613-5300	1
US ARMY MATERIEL SYS ANAL ACTVTY ATTN AMXSY CS BRADLEY APG MD 21005-5071	1
US ARMY NATICK RDEC ACTING TECHL DIR ATTN SBCN TP P BRANDLER KANSAS ST BLDG 78 NATICK MA 01760-5056	1
US ARMY OEC ATTN CSTE AEC FSE 4501 FORD AVE PARK CENTER IV ALEXANDRIA VA 22302-1458	1
US ARMY SIMULATION TRAIN & INSTRMNTN CMND ATTN AMSTI CG M MACEDONIA 12350 RESEARCH PKWY ORLANDO FL 32826-3726	1
US ARMY TACOM ARDEC ATTN AMSTA AR WEL TL BLDG 59 PHILLIPS RD PICATINNY ARSENAL NJ 07806-5000	1
US ARMY TRADOC ANAL CMND WSMR ATTN ATRC WSS R WSMR NM 88002-5502	1
NAV POSTGRADUATE SCHL DEPT OF METEOROLOGY ATTN P FREDERICKSON 1 UNIVERSITY CIR MONTEREY CA 93943-5001	1

	Copies
NAV AIR WAR CTR WPN DIV ATTN CMD 420000D C0245 A SHLANTA 1 ADMIN CIR CHINA LAKE CA 93555-6001	1
AFCCC/DOC ATTN GLAUBER 151 PATTON AVE RM 120 ASHEVILLE NC 28801-5002	1
HDQTRS AFWA/DNX 106 PEACEKEEPER DR STE 2N3 OFFUTT AFB NE 68113-4039	1
USAF ROME LAB TECH ATTN CORRIDOR W STE 262 RL SUL 26 ELECTR PKWY BLDG 106 GRIFFISS AFB NY 13441-4514	1
AIR FORCE ATTN WEATHER TECHL LIB 151 PATTON AVE RM 120 ASHEVILLE NC 28801-5002	1
COLORADO STATE UNIV DEPT OF ATMOS SCI ATTN R A PIELKE 200 WEST LAKE STREET FT COLLINS CO 80523-1371	1
DUKE UNIV PRATT SCHL OF ENGRG DEPT OF CIVIL & ENVIRON ENGRG ATTN R AVISSAR HUDSON HALL BOX 90287 DURHAM NC 27708	1
THE CITY COLLEGE OF NEW YORK DEPT OF EARTH & ATMOS SC ATTN S D GEDZELMAN J106 MARSHAK BLDG 137TH AND CONVENT AVE NEW YORK CITY NY 10031	1
UNIV OF ALABAMA AT HUNTSVILLE DEPT OF ATMOS SCI ATTN R T MCNIDER HUNTSVILLE AL 35899	1
DIRECTOR US ARMY RSRCH LAB ATTN AMSRD ARL RO EV W D BACH PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709	1

	Copies
US ARMY RSRCH LAB ATTN AMSRD ARL D J M MILLER ATTN AMSRD ARL CI OK T TECHL PUB (2 copies) ATTN AMSRD ARL CI OK TL TECHL LIB (2 copies) ATTN IMNE ALC IMS MAIL & RECORDS MGMT ATTN AMSRD ARL CI CS R MEYERS ATTN AMSRD ARL CI CS D GARVEY ATTN AMSRD ARL CI P CLARK ATTN AMSRD ARL CI CN A TUNICK ATTN AMSRD ARL SE EE Z G SZTANKAY ATTN AMSRD ARL CI ES J M NOBLE ATTN AMSRD ARL SE SA N SROUR ADELPHI MD 20783-1197	12
COMMANDING GENERAL US ARMY MATERIAL CMD ATTN AMSRD SS T 60006 TH ST STE 100 FT BELVOIR VA 22060	1
IMETS PROJECT OFFICE ATTN SFAE C3T IE MET BLDG 1622 RM 131 WSMR NM 88002-5501	1
NCSU ATTN DR JERRY DAVIS PO BOX 8208 RALEIGH NC 27650-8208	1
US ARMY FIELD ARTILLERY SCHOOL ATTN ATSF TSM TA FT SILL OK 73503-5600	1
NAVAL SURFACE WEAPONS CTR ATTN CODE G63 DAHLGREN VA 22448-5000	1
US ARMY CORPS OF ENGRS ENGR TOPOGRAPHICS LAB ATTN CETEC TR G P F KRAUSE FT BELVOIR VA 22060	1
US ARMY TOPO ENGR CTR ATTN CETEC ZC 1 FT BELVOIR VA 22060-5546	1
US ARMY RESEARCH LAB ATTN AMSRD ARL CI COMP & INFO SCI DIR WSMR NM 88002-5501	1

	Copies
US ARMY MISSILE CMND ATTN AMSMI REDSTONE ARSENAL AL 35898-5243	1
WSMR TECH LIBRARY BR ATTN STEWS IM IT WSMR NM 88002	1
US ARMY CECOM INFORMATION & INTELLIGENCE WARFARE DIRECTORATE ATTN AMSEL RD IW IP FORT MONMOUTH NJ 07703-5211	1

	Copies
DEFENSE TECHNICAL INFO CTR ATTN DTIC OA MS CORA SANTOS 8725 JOHN J KINGMAN RD STE 0944 FORT BELVOIR VA 22060-6218	1 ELEC
US ARMY RESEARCH LAB AMSRD CI OK TP TECHL LIB APG MD 21005	2
US ARMY RESEARCH LAB AMSRD ARL CI EM ATTN S KIRBY (2 hard copies) T HENMI (1 CD) WSMR NM 88002-5501	2 HCs + 1 CD
TOTAL	59 (2 HCs + 56 CDs + 1 elec)