

Steve Ansari *

STG Incorporated, Asheville, North Carolina

Stephen Del Greco

NOAA National Climatic Data Center, Asheville, North Carolina

ABSTRACT

The NCDC ingests and archives, on average, 80 terabytes of NEXRAD Radar Data and products annually. These data are in high demand globally by both the public and private sectors. As much as one terabyte of data have been accessed monthly through the NCDC radar resources web page. In an effort to provide better support to these end users, NCDC has developed visualization tools for browsing and displaying these data.

The NCDC NEXRAD Interactive Viewer and Data Exporter load NEXRAD volume scan data, known as Level-II and derived products, known as Level-III, into an OpenGIS compliant environment [6]. The applications are launched via Java Web Start and run on the client machine while accessing the data remotely from the archive at the NCDC [5]. The NEXRAD Interactive Viewer provides tools for custom data overlays, animations and basic queries. The export of images and movies is provided in multiple formats. The NEXRAD Data Exporter allows for data export in both vector polygon (Shapefile, GML, Well-Known Text) and raster (GeoTIFF, ARC/INFO ASCII Grid, HDF, NetCDF, GrADS, binary) formats [Appendix A].

1. INTRODUCTION

The NCDC Archive, commonly known as the Hierarchical Data Storage System (HDSS), currently stores 1.2 petabytes of NEXRAD Radar data from the network of 159 Weather Surveillance Radars – 1988 Doppler (WSR-88D). This amount grows at a rate of 480 gigabytes per day, with a potential increase to 6.5 terabytes per day as new radar technologies are implemented [3]. The data are acquired in near real-time with a period of record beginning in 1991 and are provided at no cost. While radar data are instrumental in climate research and severe weather analysis, they are also used in many other disciplines including hydrology, geography, biology and engineering.

2. PURPOSE

The NCDC NEXRAD archive data are in a unique binary

format. By decoding the data into OpenGIS vector features and raster grid coverages, the data become accessible to any OpenGIS compliant Application Programming Interface (API) [6]. The NEXRAD Interactive Viewer displays the data in a GeoTools-based GIS environment while the NEXRAD Data Exporter provides the export of data to multiple common formats [2]. Having NEXRAD data in multiple common scientific formats promotes interdisciplinary use of NEXRAD data. The NEXRAD Data Exporter acts as a “bridge” between the NEXRAD data and existing software. Both NEXRAD Level-II and Level-III are supported.

3. INSTALLATION

The NEXRAD Interactive Viewer and Data Exporter are both written entirely in the Java programming language. Java programs are system-independent; they may be run without recompilation on a machine that has Java installed. Java is easy to install, free and open-source [4]. The Java NEXRAD Viewer and Exporter are launched via Java Web Start. Java Web Start allows Java software applications to be deployed to the user's machine in one seamless action. When invoked, Java Web Start checks for the most current version of the Viewer and Exporter and updates are automatically downloaded if necessary [5]. To use the Viewer or Exporter, follow the instructions on the NCDC Java NEXRAD Viewer web page at the following URL: <http://www.ncdc.noaa.gov/oa/radar/jnx/index.html>.

4. FEATURES

The NEXRAD Interactive Viewer provides several display and analysis features:

1. Data overlays for states, counties, cities, interstates, airports and NEXRAD sites (Figure 1).
2. Load custom Shapefiles [1]. This allows for the addition of high-resolution local data to map.
3. Animations: View and save infinite length animations as multiple movie formats.
4. Filter NEXRAD Data and compare results to all loaded data.
5. Perform grid math operations on rasterized data (max, min, avg, sum) (Figure 2, 3).
6. Save viewer image to multiple image formats.
7. Display Level-III alphanumeric data with graphic Level-III or Level-II backgrounds (Figures 4, 5).

* Corresponding author address: Steve Ansari, National Climatic Data Center, 151 Patton Avenue, Asheville, NC 28801; e-mail: steve.ansari@noaa.gov.

8. Save and load viewer session data and settings.

The NEXRAD Data Exporter adds several analysis features:

1. Export data to Shapefile, GML and Well Known Text (WKT) vector formats and GeoTIFF, HDF, NetCDF, ARC/INFO ASCII Grid, GrADS binary and simple binary raster formats [Appendix A].
2. User selectable raster resolution.
3. Polygon reduction of vector data. This significantly reduces the size of the output vector file.

5. PROCESSING

NEXRAD data are ordered from the NCDC HDSS via the HDSS Access System (HAS) (<http://has.ncdc.noaa.gov>). A typical order takes between 15 minutes and one hour to be temporarily staged on an NCDC FTP server. The Viewer and Exporter access the data directly from the NCDC FTP server. In addition, NEXRAD data in native formats are readable from local files or any web server. The Exporter can also download ordered data to the user computer for permanent use.

NEXRAD Level-II products are decoded through the incorporation of the Java IRAS code [7]. Any elevation angle and moment (reflectivity, velocity and spectrum width) may be selected. NCDC Decoders are used to decode the NEXRAD Level-III products. The graphic NEXRAD data are decoded into OpenGIS compliant polygons with no loss of resolution [6]. Level-III Alphanumeric products (hail, tornado, storm tracking, etc.) are decoded into point or line vector formats. All geometry operations use the Vividsolutions' Java Topology Suite (JTS) [8]. The data are rasterized for efficient display. Both the raster and vector (polygon) representation of the data are available for display. The vector features and raster grid coverage are rendered using the GeoTools Java GIS API [2] (Figure 6).

6. FLEXIBILITY

The Java NEXRAD Viewer and Exporter may be run as interactive GUI programs, or as batch processes. Both programs can process data from the NCDC, the local machine or any web server. This allows for real-time analysis and image creation when used with a real-time data feed. Furthermore, the Java NEXRAD API may be used to create custom applications such as the integration into a Web Map Service (WMS). The entire project is open-source.

7. APPLICATIONS

There are many applications of the Viewer and Exporter. For a case study, the NEXRAD Data Exporter was used to convert the Level-III Storm Total Precipitation product from Hurricane Charley (August 13-14, 2004) into Shapefile format. The resulting shapefiles were loaded into ESRI ArcGIS and compared to 2000 US Census

Data (Figure 7). Age distributions were created for census blocks experiencing 4 or more inches of precipitation (Figures 8, 9).

8. FUTURE WORK

The following features and functionality will be added to future versions of both the Viewer and Exporter.

1. Additional export formats.
2. Rules-based mosaic process.
3. Additional import formats for custom display layers.
4. User graphical annotations.

9. CONCLUSION

The NEXRAD Interactive Viewer and Data Exporter provide visualization and analysis tools for the NEXRAD radar data stored in the NCDC Archive. Current Viewer/Exporter users include the Federal Emergency Management Agency (FEMA), NASA, the Department of Defense (DOD) and the National Weather Service (NWS), as well as NCDC operations.

The system independence, batch processing features and multiple export formats yield a powerful flexibility that promotes interdisciplinary use of NEXRAD Radar data in both real-time and historical analysis.

10. REFERENCES

1. ESRI, 1998: ESRI Shapefile Technical Description [Available from ESRI Website: <http://www.esri.com> <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>].
2. GeoTools Website: <http://www.geotools.org>.
3. Hall, Alan, 2005: *Current and Projected Receipt and Archive of WSR-88D Level II Data at the National Climatic Data Center.*, 22nd IEEE / 13th NASA Goddard (MSST2005) Conference on Mass Storage Systems and Technologies, Monterey CA, April 11-14, 2005.
4. Java Website: <http://www.java.com>.
5. Java Web Start Website: <http://java.sun.com/j2se/1.4.2/docs/guide/jws/Readme.html>.
6. OpenGIS Website: <http://www.opengeospatial.org/>.
7. Priegnitz, D.L., 1995: *IRAS: Software to display and analyze WSR-88D radar data.*, Eleventh International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Boston, American Meteorological Society, 197-199.
8. Vividsolutions Java Topology Suite Website: <http://www.vividsolutions.com/jts/jtshome.htm>.

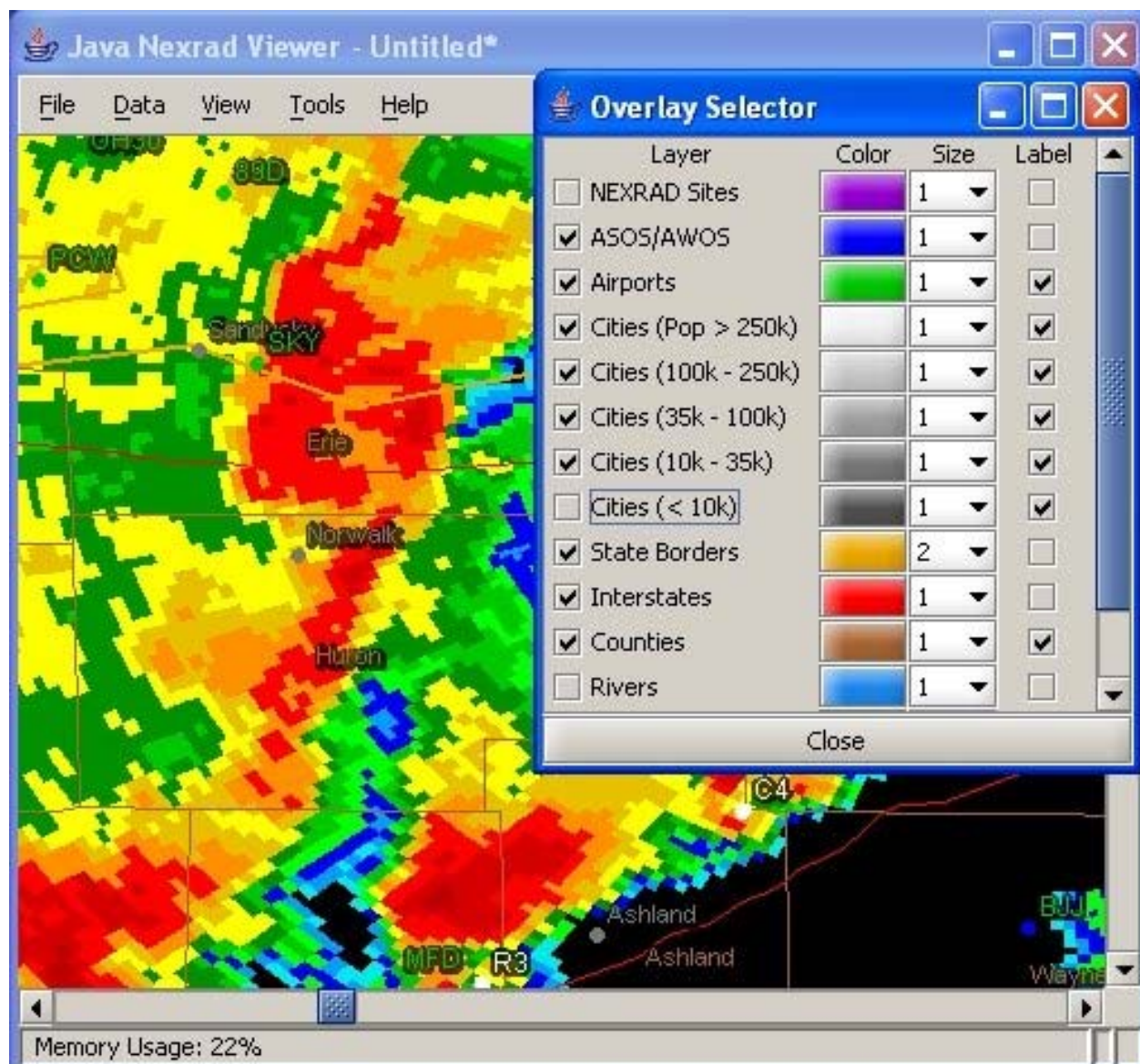


Figure 1. Map Overlays

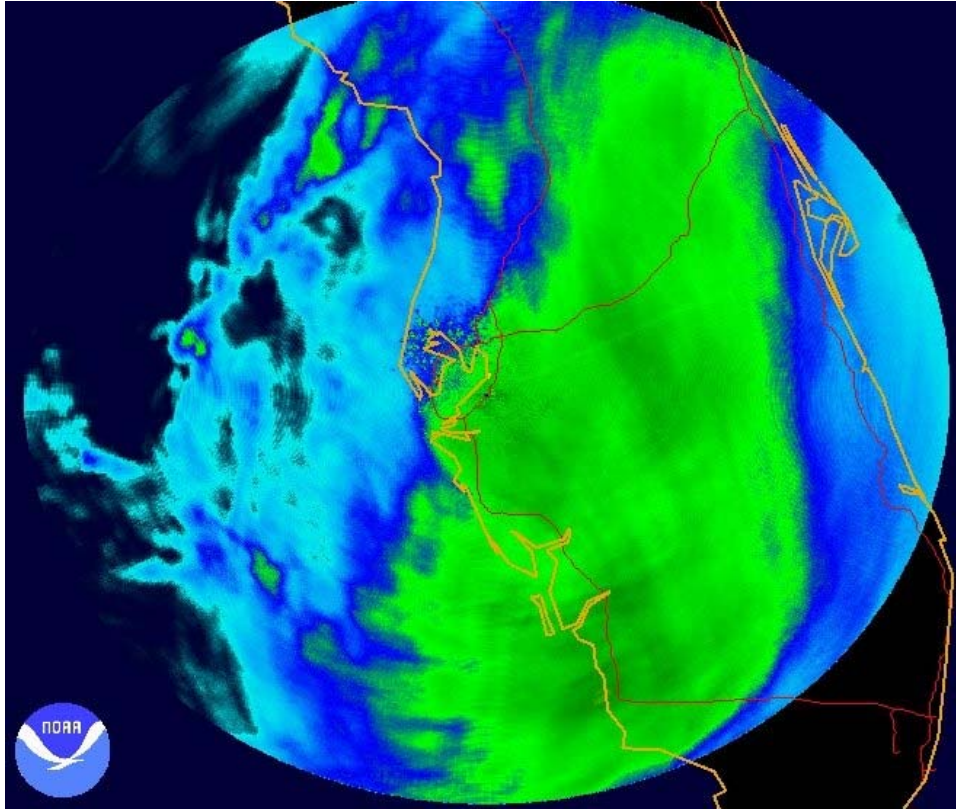


Figure 2. 7-hour average Reflectivity for Hurricane Charley (Aug. 13 2004, 17:01-23:57 UTC)

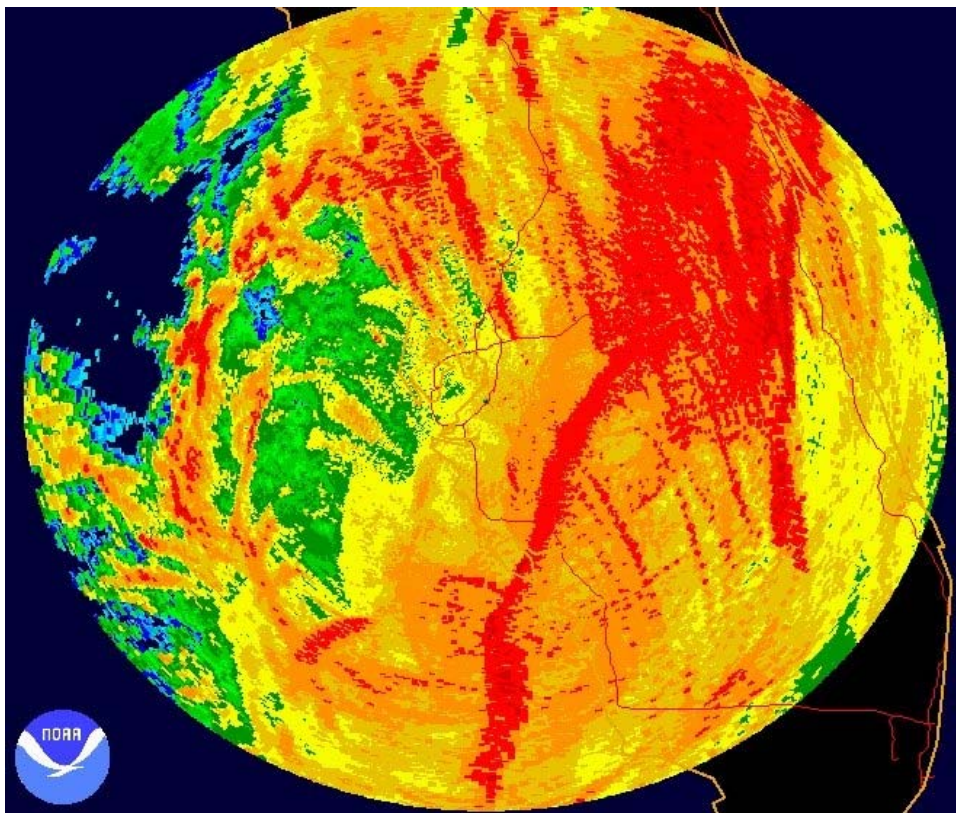


Figure 3. 7-hour maximum Reflectivity for Hurricane Charley (Aug. 13 2004, 17:01-23:57 UTC)

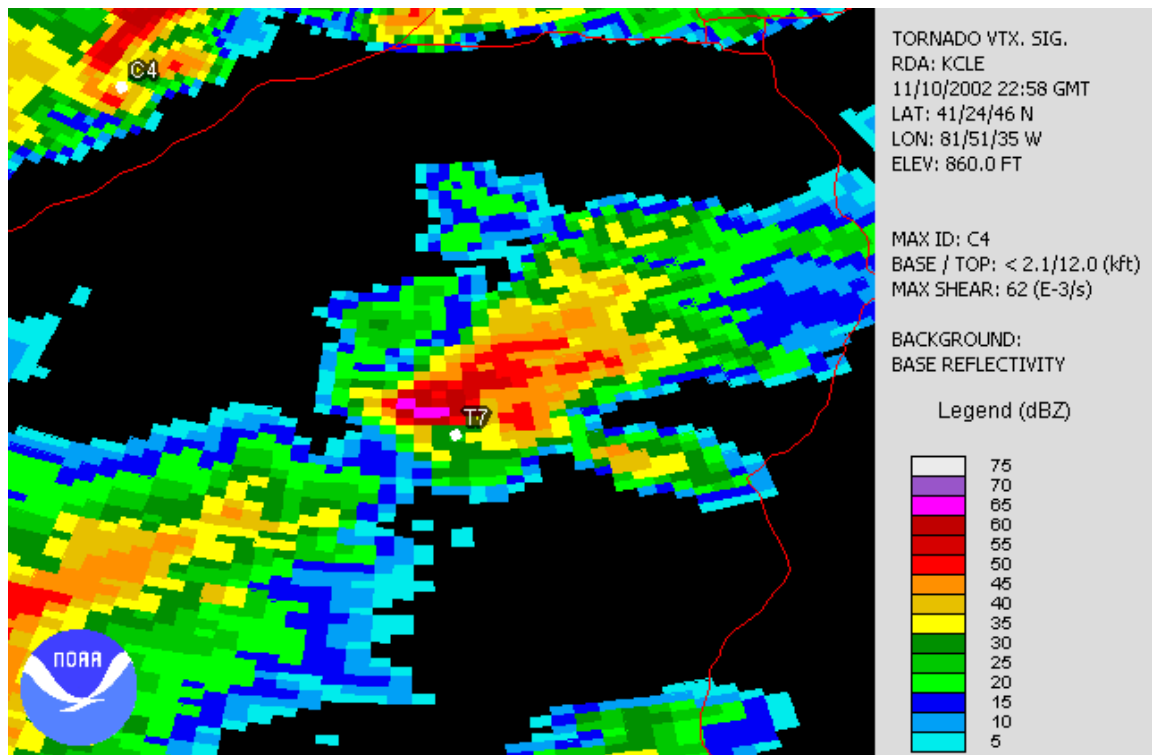


Figure 4. Tornado Vortex Signature with Base Reflectivity

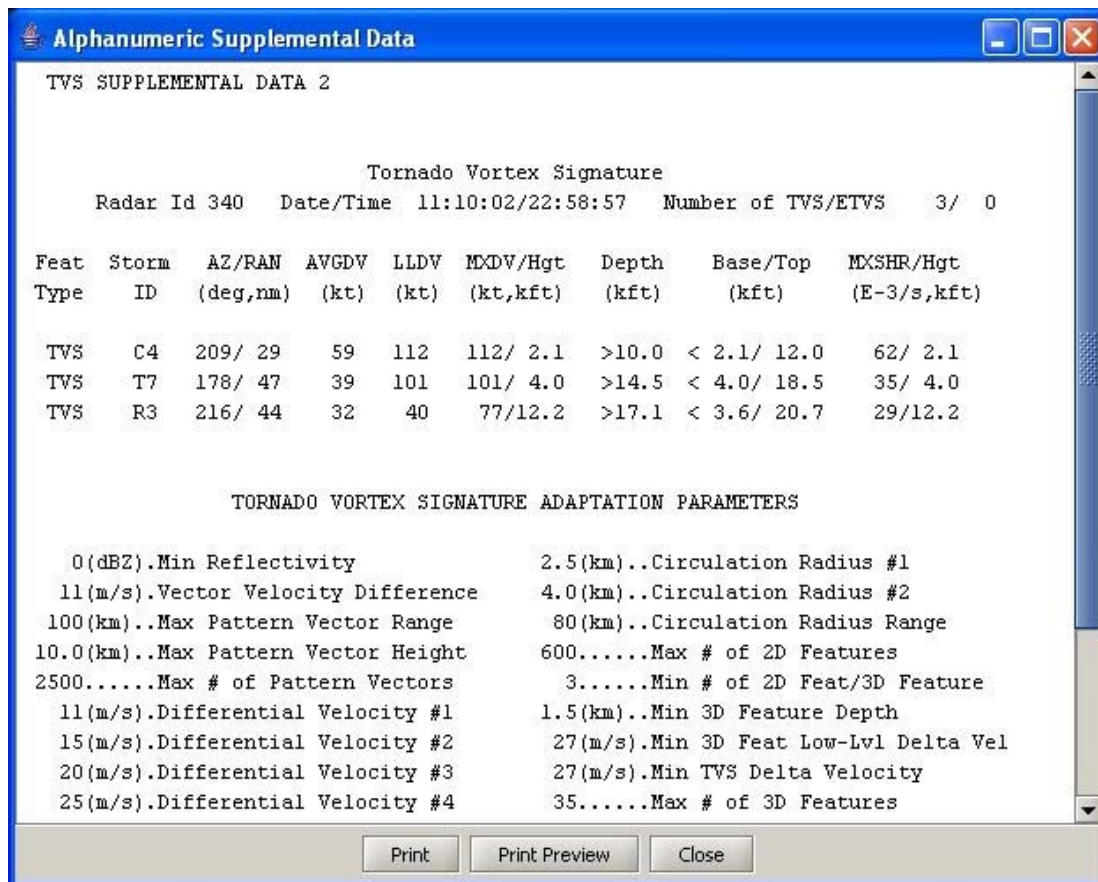


Figure 5. Tornado Supplemental Data

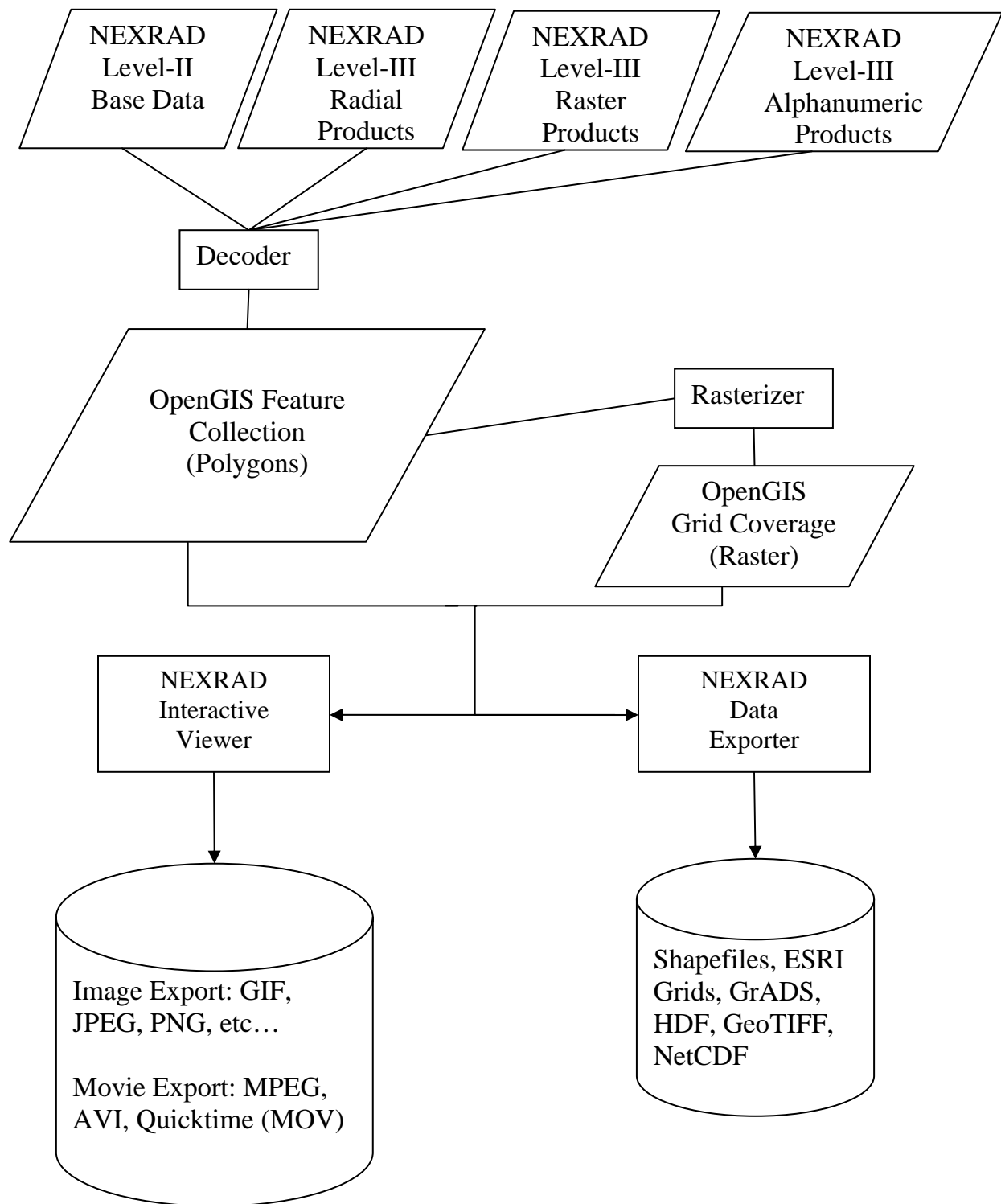


Figure 6. Java NEXRAD Flowchart

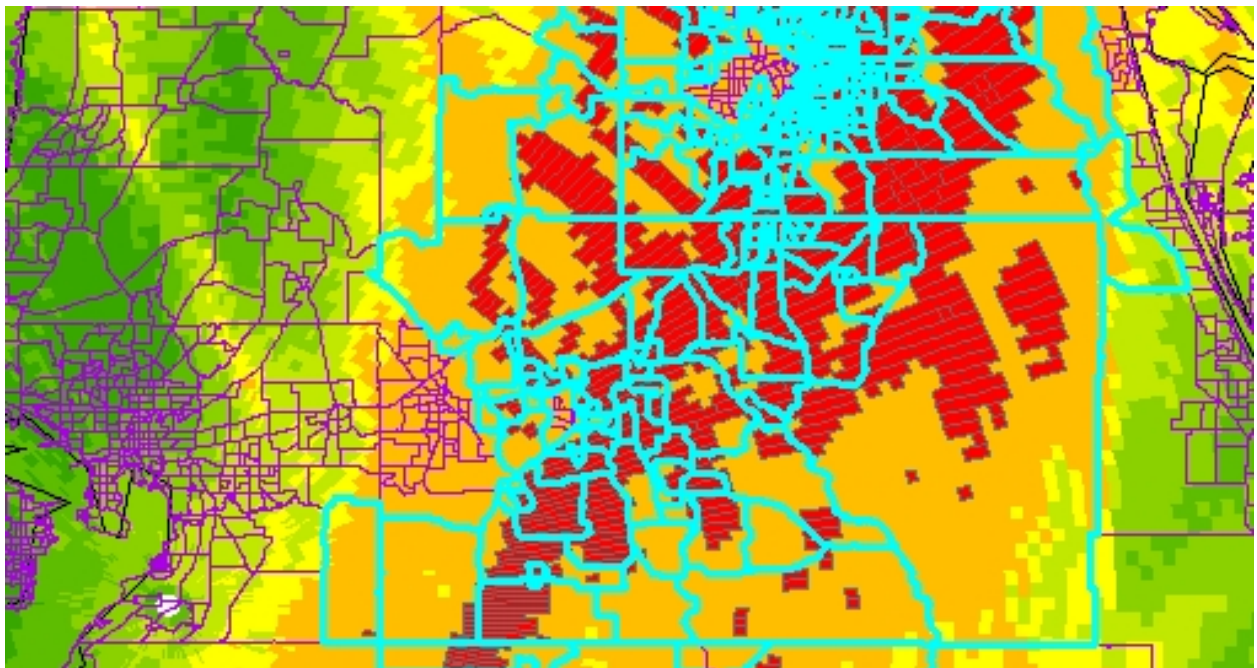


Figure 7. Census regions with ≥ 4.0 in precipitation during Hurricane Charley

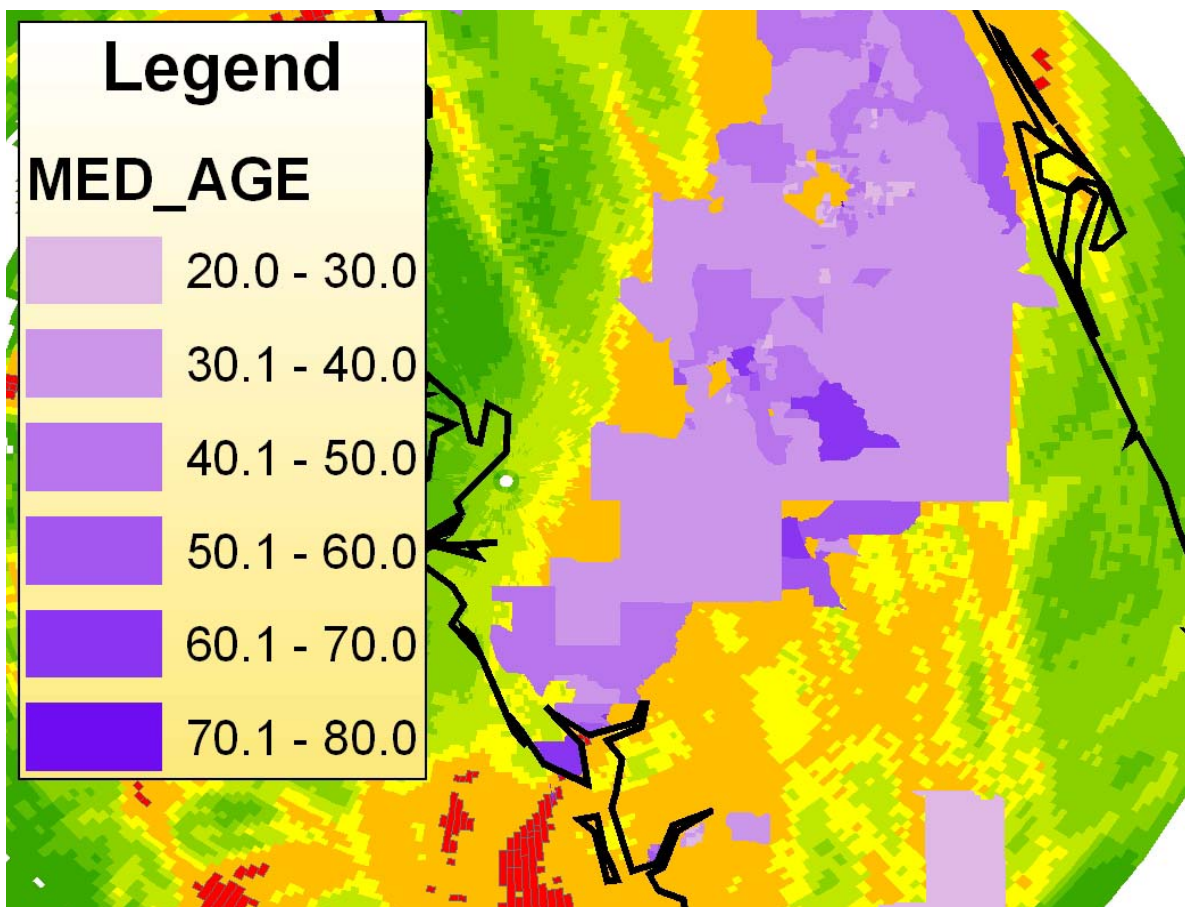


Figure 8. Age distribution map for affected regions (≥ 4.0 inches precipitation) during Hurricane Charley

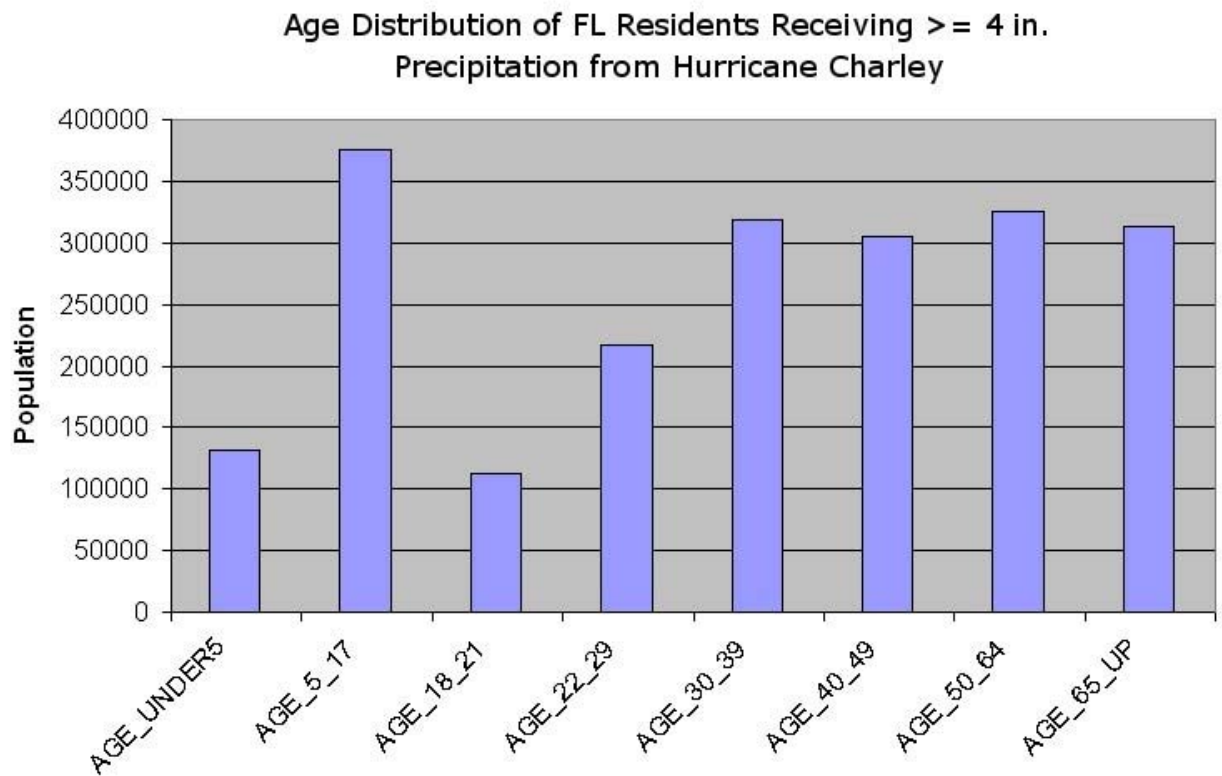


Figure 9. Age distribution graph for affected regions (≥ 4.0 inches precipitation) during Hurricane Charley

APPENDIX A

1. ESRI Shapefile [1]: "A shapefile stores nontopological geometry and attribute information for the spatial features in a data set. The geometry for a feature is stored as a shape comprising a set of vector coordinates." For more information: <http://www.esri.com>
2. Geography Markup Language (GML): "Geography Markup Language is an XML grammar written in XML Schema for the modeling, transport, and storage of geographic information." For more information: <http://www.opengis.net/gml/>
3. Well-Known Text (WKT): An ASCII text representation of geometry data. Defined in the OpenGIS Consortium "Simple Features for SQL" specification. For more information: http://dev.mysql.com/doc/mysql/en/GIS_WKT_format.html or <http://publib.boulder.ibm.com/infocenter/db2help/index.jsp?topic=/com.ibm.db2.udb.doc/opt/rsbp4120.htm>
4. GeoTIFF: "GeoTIFF represents an effort by over 160 different remote sensing, GIS, cartographic, and surveying related companies and organizations to establish a TIFF based interchange format for georeferenced raster imagery." For more information: <http://www.remotesensing.org/geotiff/geotiff.html>
5. ARC/INFO ASCII Grid: Simple ASCII text format for raster grids with simple header information embedded in the beginning of the file. For more information: <http://www.geotools.org/ArcInfo+ASCII+Grid+format>
6. Hierarchical Data Format (HDF): "At its lowest level, HDF is a physical file format for storing scientific data. At its highest level, HDF is a collection of utilities and applications for manipulating, viewing, and analyzing data in HDF files. Between these levels, HDF is a software library that provides high-level APIs and a low-level data interface." For more information: <http://hdf.ncsa.uiuc.edu/>
7. NetCDF (network Common Data Form): "NetCDF (network Common Data Form) is an interface for array-oriented data access and a freely-distributed collection of software libraries for C, Fortran, C++, Java, and perl that provide implementations of the interface. The netCDF software was developed by Glenn Davis, Russ Rew, Steve Emmerson, John Caron, and Harvey Davies at the Unidata Program Center in Boulder, Colorado, and augmented by contributions from other netCDF users. The netCDF libraries define a machine-independent format for representing scientific data. Together, the interface, libraries, and format support the creation, access, and sharing of scientific data." For more information: <http://my.unidata.ucar.edu/content/software/netcdf/index.html>
8. GrADS (Grid Analysis and Display System) Binary: Simple floating point binary grid file with additional GrADS header file containing grid specifications. For more information: <http://grads.iges.org/grads/grads.html>
9. Simple Binary: Floating point binary grid file with ARC/INFO header file containing grid specifications.