NASA/TM—20220007001

Air Traffic Management TestBed: Weather Visualization using Map Tiles

Chok Fung Lai
Ames Research Center, Moffett Field, California

May 2022

NASA STI Program ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Phone the NASA STI Information Desk at 757-864-9658

- Write to:
  NASA STI Information Desk
  Mail Stop 148
  NASA Langley Research Center
  Hampton, VA 23681-2199

NASA/TM—20220007001

Air Traffic Management TestBed: Weather Visualization using Map Tiles

Chok Fung Lai
Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, CA 94035-1000

May 2022

Acknowledgments

The author would like to thank not only the other Air Traffic Management TestBed team members, Alan Lee, Phu Huynh, Huu Huynh, Jimmy Nguyen, and Yun Zheng, for their contributions to the design and development of the weather data visualization capabilities, but also the NASA Sherlock Air Traffic Management Data Warehouse team for providing the technical support of accessing the weather data files. The author would also like to thank Yun Zheng, Dr. Todd Lauderdale, Gregory Wong, and Sandra Lozito for reviewing this technical memorandum.

This report is available in electronic form at

https://ntrs.nasa.gov/

Abstract

Running realistic simulations and developing algorithms for decision support tools usually require weather data access. This document describes the weather data service capability implemented in the Air Traffic Management (ATM) TestBed. The capability provides ways for querying weather data from the weather products provided by the National Aeronautics and Space Administration's (NASA) Sherlock ATM Data Warehouse. Depending on use cases, weather data can be queried using a Java application programming interface, web data access, and map tile image access. Map tile images are generated using triangular meshes obtained using the Matching Squares contour algorithm. An analysis determines a grid size that will produce both high image quality and fast image rendering. In addition, the visualization of weather data as map tiles on a TestBed visualization tool called Traffic Viewer is presented.

This page intentionally left blank.

Table of Content

1. Nomenclature 7
2. Introduction 8
3. Weather Data Formats 8
4. Map Tiles 10
5. Weather Visualization 13
   5.1. Data Request URLs 14
   5.2. Tile Request URLs 18
   5.3. Traffic Viewer 18
6. Concluding Remarks 21
7. References 21

List of Figures

Figure 3.1. From Weather Grids to Data Cube 9
Figure 4.1. Contouring weather data: (a) mesh and (b) image tile 12
Figure 4.2. Grid Size, Rendering Time, and Tile Image 12
Figure 4.3. Rendering Time and Grid Size 13
Figure 5.1. Cloud Coverage Tile Images 17
Figure 5.2. Cloud Coverage Tile Images with Applied Functions 18
Figure 5.3. CWAM Polygons on 2019-07-06 at 20:30Z 20
Figure 5.4. CWAM Polygons on 2019-09-01 at 20:30Z 20
Figure 5.5. Combined CWAM Polygons from 2019-07-06 and 2019-09-01 at 20:30Z 21

List of Listings

Listing 5.1. Running the Weather Tile Image Server 14
Listing 5.2. Wind Query Response 15
Listing 5.3. CWAM Weather Provider Properties 19

List of Tables

Table 1.1. Units of Measurement 7
Table 3.1. Shapes of Data Grids 9
Table 4.1. Map Tile URL Parameters 10
Table 4.2. Weather Values at Neighboring Pixels of Tile (8, 12) at Zoom Level 5 11
Table 5.1. Weather Data Request URL Parameters 14
Table 5.2. Supported Dataset Parameter Values 15
Table 5.3. Supported Resolution Parameter Values 15
Table 5.4. Supported Edition Parameter Values 16
Table 5.5. Supported Type Parameter Values 16
Table 5.6. Supported Function Parameter Values 17
Table 5.7. Special Date Time Values 19

This page intentionally left blank.

1. Nomenclature

This section lists all the notations and symbols used in this technical memorandum.

`[a,b)` = Range of values between `a` (inclusive) and `b` (exclusive).
`(a,b]` = Range of values between `a` (exclusive) and `b` (inclusive).
`[a,b]` = Range of values between `a` (inclusive) and `b` (inclusive).
<x> = Parametric field `x` in a Uniform Resource Locator.
$\lceil x \rceil$ = Ceiling value of `x`.
$\lfloor x \rfloor$ = Floor value of `x`.
`T` = Time axis.
t, t_1, t_2 = Time values (either a timestamp or a coordinate value).
v, v_1, v_2 = Weather data values.
`X, Y, Z` = Grid coordinates axes. Both `X` and `Y` are horizontal, while `Z` is vertical.
`x, y, z` = Grid coordinates along X-, Y-, and Z-axes; or tile coordinates and zoom level.

Table 1.1 lists the units of measurement used in the weather data models, query, and result.

Table 1.1. Units of Measurement

Symbol	Unit	Unit Of	Description
`%`	Percentage	-	The dimensionless unit of a fraction of 100. One percentage equals one-hundredth, i.e., $1\% = 1/100$.
`deg`	Degree	Angle	For latitudes, values are in World Geodetic System 1984 (WGS84) reference coordinate system [1]. Latitude values are between -90 and +90, inclusive. Positive values are north of the equator (N), and negative values are south of the equator (S). For longitudes, values are in the WGS84 reference coordinate system. Longitude values are between -180 and +180, inclusive. Positive values are east of the prime meridian (E), and negative values are west of the prime meridian (W).
`ft`	Foot	Height	The unit of vertical distance in air navigation.
`km`	Kilometer	Length	The unit of horizontal distance. One kilometer equals 1,000 meters.
`kt`	Knots	Speed	The unit of speed. One knot equals one nautical mile per hour.
`ms`	Millisecond	Time	The unit of time. One millisecond equals one-thousandth of a second, i.e., $1ms = 1/1{,}000\ s$. Time is measured by the number of milliseconds elapsed since 1 January 1970 00:00:00 GMT. Duration is measured by taking the difference between two timestamps.
`mps`	Meters per second	Speed	The unit of speed and velocity.
`px`	Pixel	Image dimension	The unit of image dimension. One pixel, or picture element, represents a single point in an image to be displayed on a monitor.

2. Introduction

Weather plays an important role in daily air traffic management operations [2, 3]. Creating scenarios, developing algorithms, prototyping decision support tools, and running simulations for weather impact and avoidance requires accessing, processing, and visualizing weather data to identify interesting regions of airspace [4, 5, 6, 7].

This document describes the weather data service capability implemented in the Air Traffic Management (ATM) TestBed. For algorithm prototyping and development, the ATM TestBed, or TestBed in short, provides a simple and easy capability to connect high-fidelity simulations for supporting the National Aeronautics and Space Administration (NASA) and community research [8]. Weather support was developed in TestBed's initial build and has been enhanced in subsequent builds:

- In Build 0.9a, a weather scenario generation capability was added to query weather information with four-dimensional parameters: time and geo-coordinates (latitude, longitude, and altitude).
- In Build 2.0a, a weather image tile server was added to support weather visualization on map tiles.
- In Build 2.0.2, the weather image tile server was extended to support weather data querying using Uniform Resource Locators (URLs) [9].
- In Build 2.3, the weather query supports multiple timestamps, and a value function is applied to aggregate the weather data values to a final value.

Section 3 briefly describes formats of the weather data files that are stored in Sherlock and supported in the TestBed. Section 4 presents basics of map tiles and the contouring algorithm used by the TestBed to generate weather tile images. Finally, weather data visualization on a two-dimensional visualization tool called Traffic Viewer [10] is detailed in Section 5.

3. Weather Data Formats

For weather data sources, the NASA's Sherlock ATM Data Warehouse, or Sherlock, provides an extensive list of data formats from the Federal Aviation Administration (FAA) and the National Oceanic and Atmospheric Administration (NOAA) [11] for supporting research. Sherlock includes a database with tens of terabytes of data storage capacity and a web user interface for querying and downloading data files. Sherlock provides weather products including Corridor Integrated Weather Service (CIWS) [12], Convective Weather Avoidance Model (CWAM) [13], and Rapid Refresh (RR) and Rapid Update Cycle (RUC) Weather Forecast [14, 15] for air traffic management researchers. These products use multidimensional data formats like network Common Data Form (netCDF) [16], GRIdded Binary (GRIB) [17], and Hierarchical Data Format (HDF) [18]. A Thematic Real-time Environmental Distributed Data Services (THREDDS) data server [19] is running on Sherlock to serve continental-scale and constantly updated weather data files. As a result, the weather data files can also be accessed via URLs specified by the THREDDS data server.

In Sherlock, several weather products store current and forecast weather data. File paths of the same weather product follow certain patterns and naming conventions. A file path comprises segments including a dataset, an optional resolution, an optional edition, a date, a time, an optional forecast hour, and a file extension. Given a weather data query, a specific data file can be located and downloaded from the THREDDS data server. Here are some example paths of the weather data files stored in Sherlock:

- `ciws.EchoTop.Netcdf4.1km/2019/12/19/ciws.EchoTop.20191219T000000Z.nc`

  dataset resolution date time file extension
- `cwamModels/2019/07/06/2019_07_06_20_30_GMT.Forecast.h5.CWAM.h5.bz2`

  dataset date time file extension

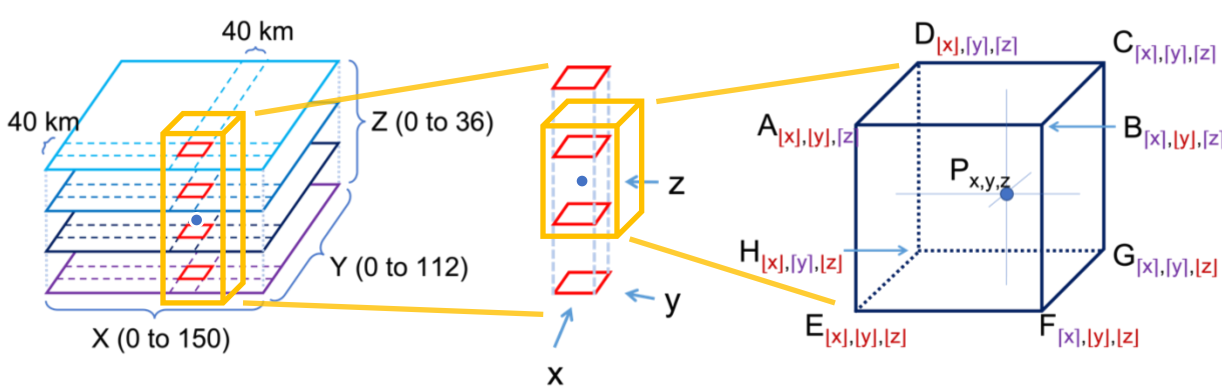
```
• RR_CONUS_40km_Grib2/2015/06/22/rr.rap.t19z.awp236pgrbf01.20150622.grib2
  dataset resolution edition date      time        forecast      file
                                                     hour       extension
• RUC_CONUS_13km_Grib1/2012/03/24/ruc.fh.0001_tl.press_gr.us13km.23Z.20120324.grib1
    dataset resolution edition date     forecast                 time          file
                                          hour                             extension
```

Though the data models use netCDF, GRIB, and HDF storage formats, they can be accessed using the Unidata NetCDF library [16]. Table 3.1 lists the horizontal resolution distance and the four-dimensional shape of the data grids: (1) T, the time axis; (2) Z, the vertical height axis; (3) Y, the horizontal latitude axis; and (4) X, the horizontal longitude axis. The T, Z, Y, and Z column values represent the numbers of grid cells along the time, vertical height, horizontal latitude, and horizontal longitude axes, respectively. The CWAM data model is hierarchical based. The library can access the data including altitudes, thresholds, and polygon vertices. The TestBed provides a weather service component for accessing these data models using weather service application programming interfaces (APIs) and web URLs.

Table 3.1. Shapes of Data Grids

Product	Resolution	T	Z	Y	X
CIWS	1 km	24	1	3,520	5,120
RR, RUC	13 km	1	37	335	451
RR, RUC	40 km	1	37	131	151

Certain Sherlock weather products, including CIWS, RR, and RUC, use grid-based data models to store values in the Continental United States. According to the file naming conventions, both RR and RUC use Lambert conformal conic projections. Vertical data are recorded depending on the types of the data. For example, wind speed gusts use ground or water surface level, temperatures use pressure level at isobaric surface, and pressures represent height level above ground. The weather data models, queries, and results in the TestBed are constructed using the Java programming language [20] and their data types follow the Java Language Specification [21].



(a) 40-km grids *(b) Grid column* *(c) Data cube*

Figure 3.1. From Weather Grids to Data Cube

Figure 3.1 shows notional weather grid coordinates defined in a weather data file with the 40-km horizontal resolution. In Figure 3.1(a), the blue point, shown inside a yellow cuboid, represents a point of interest in a weather query. The number of grid cells defined in the weather data file is 151 (X) ×113 (Y) × 37 (Z). Note that arrays in the Java programming language use 0-based indices. Thus, the ranges of the X-, Y-, and Z-axis indices are integers in `[0,150]`, `[0,112]`, and `[0,36]`, respectively. Figure 3.1(b) zooms in to the vertical grid column containing the blue point where x, y, and z are the numeric (not necessary integers) coordinates along the X-, Y-, and Z-axes, respectively. Figure 3.1(c) further zooms in to the data cube covering the top and bottom grid surfaces. Each grid surface has four corners, and each corner represents a grid point defined in the weather data file. In this figure, red and purple axis-coordinates indicate floor and ceiling values, respectively.

To support smooth integration functions in trajectory prediction, linear interpolations are used to make sure weather data values obtained from the weather queries are continuous. Given a point P with coordinates x, y, and z, one can find the data cube containing the point P. The eight corners, A to H, of the data cube are the grid points defined in the weather data file. The weather data value of P is calculated based on the eight corner values by using the following linear interpolations:

1. $P_{x,y,\lceil z \rceil}$: Use the X- and Y-axis values at the top four corners, A, B, C, and D, by linearly projecting P onto the top surface with Z-axis value $\lceil z \rceil$.
2. $P_{x,y,\lfloor z \rfloor}$: Use the X- and Y-axis values at the bottom four corners, E, F, G, and H, by linearly projecting P onto the bottom surface with Z-axis value $\lfloor z \rfloor$.
3. $P_{x,y,z}$: Calculate the weather data value by linearly projecting the two values, $P_{x,y,\lceil z \rceil}$ and $P_{x,y,\lfloor z \rfloor}$, obtained in steps 1 and 2 above, along the Z-axis.

For weather forecasts, the time, t, defined in a weather query is between two data files with forecast time values, t_1 and t_2, respectively. In this case, another linear interpolation is applied to the two calculated forecast data values, $v_{t1} = P_{x,y,z}(t_1)$ and $v_{t2} = P_{x,y,z}(t_2)$. Thus, the weather data value, v_t, at time t is defined as $v_t = v_{t1} + \frac{t-t_1}{t_2-t_1} \cdot (v_{t2} - v_{t1})$ where $t_1 \le t < t_2$.

On the other hand, the CWAM weather product in Sherlock uses HDF instead of GRIB. The CWAM data file stores vertices of weather polygons with respect to their deviation probability (or percentage) thresholds. In this case, the TestBed weather query will return the threshold value by finding a polygon containing the geo-location point P.

4. Map Tiles

Map tiles are widely used in map applications. Each tile is an image with pixel dimensions of 256 × 256. The spatial location of a tile is usually referenced by three parameters:

a. 0-based zoom level, z, between 0 and 23, inclusive.
b. 0-based x-coordinate of the tile, between 0 and $2^z - 1$, inclusive.
c. 0-based y-coordinate of the tile, between 0 and $2^z - 1$, inclusive.

A common format of general map tile URLs on a web server serving Hypertext Transfer Protocol (HTTP) or Hypertext Transfer Protocol Secure (HTTPS) appears as [22]:

```
http[s]://<host>[:<port>]/<type>/tile/<zoom>/<tileY>/<tileX>
```

where the parametric fields are listed in Table 4.1.

Table 4.1. Map Tile URL Parameters

Field	Type	Description	Example
`<host>`	`String`	Host name of the server.	localhost
`<port>`	`int`	Port number of the server. The default values for the HTTP and HTTPS are 80 and 443, respectively.	7890

<type>	String	Type of the map tile images. It may contain more than one segment. This allows the same server to be used for serving different image tiles such as satellite images and topographic images.	services/topo
<zoom>	int	0-based zoom level.	5
<tileY>	int	0-based y-coordinate of the tile.	12
<tileX>	int	0-based x-coordinate of the tile.	8

A tile image has 65,536 pixels (256 px × 256 px). Since each pixel represents an aggregated value of a specific geo-location region, it would be compute-intensive and time-consuming to perform about 66 thousand weather queries on the server side to render a single image. In addition, multiple tile images would need to be served to fill a view port of a map application. For a view port with pixel dimensions of 1024 × 768, the number of tile images to be served is between 12 and 20[1]. The lower bound occurs when the top-left corner of the tile image is located at the top-left corner of the view port; the upper bound occurs when the top-left corner of the tile image is outside the view port. Weather values at neighboring pixels (or coordinates) are similar, i.e., within a small threshold. This can be illustrated by querying wind magnitudes using the RR 40-km resolution at four close geo-locations at 30,000-ft altitude on 22 June 2015 at 21:30Z. Table 4.2 lists the wind magnitudes, in knots, of Pixel P1 and its three neighboring pixels, P2, P3, and P4. Note that the wind magnitude values of the four pixels are between 49.45 and 49.74[2] knots. Each pixel of the tile image, Δp, represents $1/_{256}$-th units of distance in tile coordinates.

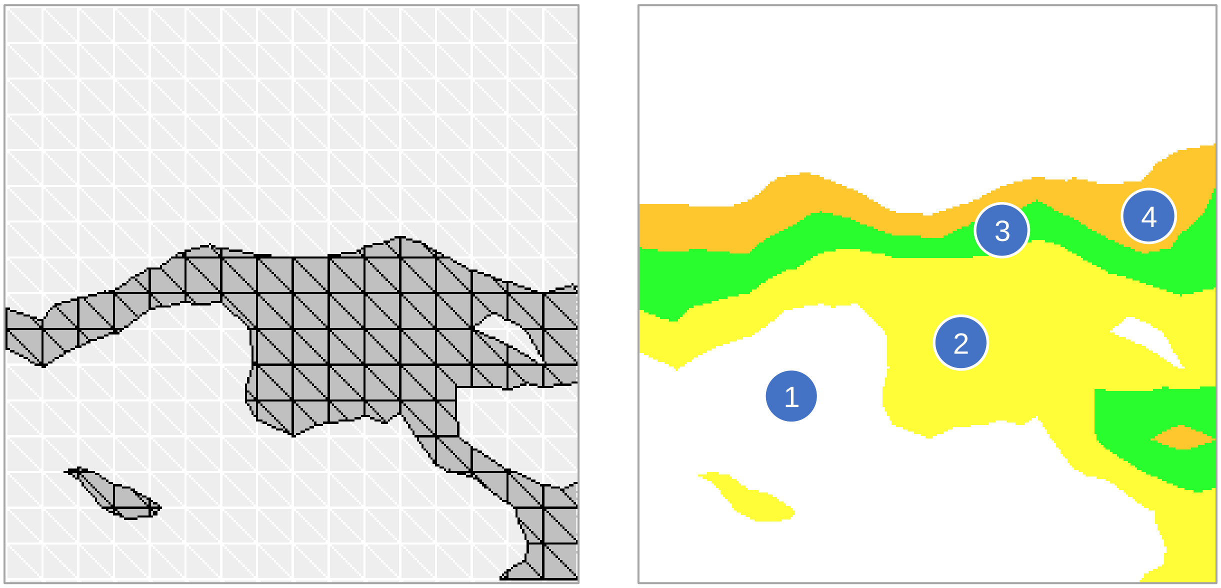
Table 4.2. Weather Values at Neighboring Pixels of Tile (8, 12) at Zoom Level 5

Pixel	**Tile X**	**Tile Y**	**WGS84 Coordinates (latitude (deg.), longitude (deg.))**	**Wind Magnitude (knots)**
P1	8	12	(40.979897, -90.0)	49.699486
P2	8 + Δp	12	(40.979897, -89.956055)	49.733738 (max)
P3	8	12 + Δp	(40.946712, -90.0)	49.457653 (min)
P4	8 + Δp	12 + Δp	(40.946712, -89.956055)	49.497086

To speed up the tile image rendering process TestBed uses the Matching Squares contouring algorithm with triangular meshes [23]. The algorithm generates triangular meshes for values in a specified range called *isoband*.

[1] Lower bound = $\frac{1024}{256} \times \frac{768}{256}$ = 4 × 3 = 12. Upper bound = $\left(\frac{1024}{256} + 1\right) \times \left(\frac{768}{256} + 1\right)$ = 5 × 4 = 20.

[2] Rounding down/up the minimum/maximum values, respectively, to two decimal places.



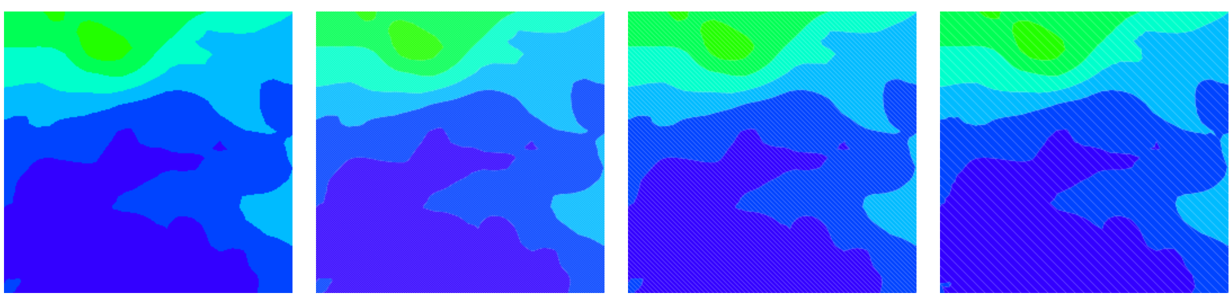
(a) Meshes of values in `[10,20)` (b) Isobands of values in `[0,40)`

Figure 4.1. Contouring weather data: (a) mesh and (b) image tile

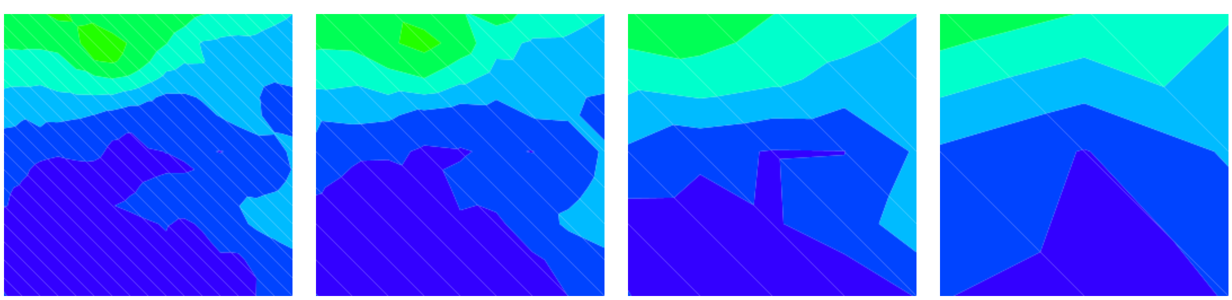
Figure 4.1 shows two example images generated by the algorithm. Assuming weather values are between 0 (inclusive) and 40 (exclusive), i.e., `[0,40)`. Figure 4.1(a) depicts the triangular meshes for the isobands with values in `[10,20)`. Figure 4.1(b) shows four isobands:

1. White regions with values in `[0,10)`,
2. Yellow regions with values in `[10,20)`,
3. Green regions with values in `[20,30)`, and
4. Orange regions with values in `[30,40)`.

Note that the meshes shown on the left also represent the yellow regions shown on the right.



(a) 1 px, 878 ms *(b) 2 px, 341 ms* *(c) 4 px, 131 ms* *(d) 8 px, 66 ms*



(e) 16 px, 26 ms *(f) 32 px, 25 ms* *(c) 64 px, 22 ms* *(d) 128 px, 19 ms*

Figure 4.2. Grid Size, Rendering Time, and Tile Image

Using a larger grid will reduce the number of weather data queries, thereby speeding up the contouring algorithm and tile image generation. However, using a larger grid will also reduce the tile image quality. Eight tile images based on different grid sizes and their rendering times are depicted in Figure 4.2. The eight tile images represent wind magnitudes based on the one-hour RR forecast data, 40-km horizontal resolution, and second edition GRIB at 30,000-ft altitude on 22 June 2015 at 19:00 Z. The tile coordinates are (x=8, y=12) at zoom level z=5. Dark blue regions represent wind magnitudes in (0,10] knots; light green regions represent wind magnitudes in (60,70] knots. The URL of the weather map tiles in this figure is http://<host>:<port>/rr/40km/2/wind/2015-06-22T19:00Z/1:0/30000/tile/5/12/8 (see Section 5.1). Based on visual observation, the images indicate that using four-pixel grids provide a good compromise between computation time and image quality.

Figure 4.3 shows a semi-logarithmic plot indicating an inversely proportional relationship of the image tile's rendering time, in milliseconds, versus its grid size, in pixels. The horizontal axis uses the base-2 logarithm scale ranging from $2^0 = 1$ to $2^7 = 128$. Besides the one-pixel grid size case ($2^0 = 1$), the contouring algorithm is applied when rendering the image tile.

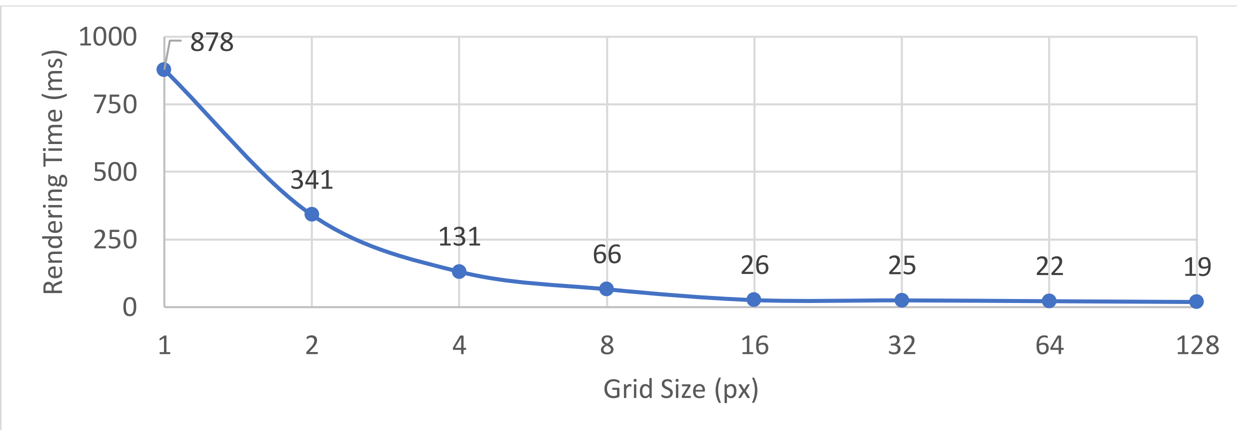


Figure 4.3. Rendering Time and Grid Size

5. Weather Visualization

TestBed provides Java APIs for querying weather data. In addition, two approaches may be used to display weather information on a map visualization tool:

1. Publish weather data from a weather publisher to the visualization tool, and the visualization tool renders the weather polygons. This approach allows other projects to access the raw weather data defined in the data exchange messages. However, individual subscribers need to parse and interpret the weather data.
2. Publish map tile images from a weather server to the visualization tool, and the visualization tool displays the map tile images. This approach seamlessly integrates with the existing map applications that can display map tile images. In addition, less data needs to be transferred between the publisher and the subscriber. However, individual subscribers will not have access to the weather data.

This section presents a hybrid approach by implementing a weather image server that can serve both weather data and map tiles. Depending on the use cases, algorithms and decision support tools can access the numeric values, while map visualization tools can access the tile images. Listing 5.1 lists the console commands to build and run the TestBed Weather Tile Image Server on three major operating systems. Press <ctrl> <c> on the console stops the server.

Listing 5.1. Running the Weather Tile Image Server

Platform	Command

Linux, macOS	`$ cd $SNTB_HOME/TestBedCore/WeatherSupport/WeatherImageServer/`
	`$ gradle shadowJar`
	`$ ./bin/run_tile_image_server.sh -port 7890`

Windows	`> cd %SNTB_HOME%\TestBedCore\WeatherSupport\WeatherImageServer\`
	`> gradle shadowJar`
	`> .\bin\run_tile_image_server.bat -port 7890`

5.1. Data Request URLs

The interpolated numeric values from the weather data files can be obtained by accessing the data request URLs, which have the following parametrized format:

```
http://<host>:<port>/<dataset>/<resolution>/<edition>/<type>/<time>/
<forecast>/<altFt>/data/<latDeg>/<lonDeg>[?apply=<func>]
```

where the parametric fields are listed in Table 5.1 through Table 5.6.

Table 5.1. Weather Data Request URL Parameters

Field	Type	Description	Example
`<host>`	`String`	Host name of the server.	localhost
`<port>`	`int`	Port number of the server.	7890
`<dataset>`	`String`	Dataset of the weather product used by the Sherlock THREDDS data server. See Table 5.2 for the supported values.	ciws
`<resolution>`	`String`	Horizontal resolution of the dataset. See Table 5.3 for the supported values.	1km
`<edition>`	`String`	Edition of the dataset. See Table 5.4 for the supported values.	1
`<type>`	`String`	Type of the request. See Table 5.5 for the supported values.	echo-top
`<time>`	`String`	Current simulation time, either a date/time string or a long value indicating the number of milliseconds elapsed since 1 January 1970, 00:00:00 GMT: • `yyyy-MM-dd['T'HH[:mm[:ss[.SSS]]]'Z']` • long (in ms) A value of zero (0) indicates the time of the weather image server. Values may be comma-separated values for combining weather data from multiple days.	2019-07-06T21:30Z 0,2019-07-06T21:30Z 1435008600000
`<forecast>`	`String`	Forecast duration, in `HH:MM` format. Values are usually between `00:00` (no forecast) and `06:00` (six hours).	1:00
`<altFt>`	`float`	Altitude, in feet, of the geo-location. Note that this value will not be used for weather fields supporting two-dimensional areas without vertical information, e.g., total cloud coverage.	30000

`<latDeg>`	`double`	Latitude of the geo-location, in degrees.	37.615223
`<lonDeg>`	`double`	Longitude of the geo-location, in degrees.	-122.389977
`<func>`	`String`	Function to be applied to the weather values. The default value is `max` whenever the parameter is not specified. See Table 5.6 for the supported values.	max

The following is an example URL to access the wind data at 12,000 ft above San Francisco International Airport's Terminal 2 on 22 June 2019 at 18:30:42 Z. The server response is listed in Listing 5.2.

```
http://localhost:7890/rr/40km/2/wind/2019-06-22T18:30:42Z/00:00/12000/
data/37.617362/-122.381540
```

Listing 5.2. Wind Query Response

```
u_mps=-2.4608707440823263;v_mps=-7.569231505072198
```

TestBed's weather image server currently supports four datasets, and their parameter values are listed in Table 5.2. Additional datasets can be added in the future by introducing a new Java subpackage in the package `gov.nasa.sntb.weathersupport.weatherservice.thredds` and including the following subclasses to the subpackage.

- `DatasetPath`: Represents a dataset path of the new weather data file.
- `DatasetPathMap`: Stores mappings of time and dataset paths.
- `DatasetPathPatterns`: Defines regular expression patterns for the new dataset paths.
- `Source`: Represents a source of the weather product data files.
- `ValueGrid`: Calculates grid values using provided linear interpolation APIs.
- `WeatherGrid`: Provides weather querying APIs for the new weather data values.
- `WeatherProvider`: Provides weather grid and field testing for a weather product.

Table 5.2. Supported Dataset Parameter Values

Value	**Description**
`ciws`	Corridor Integrated Weather System
`cwam`	Convective Weather Avoidance Model
`rr`	Rapid Refresh
`ruc`	Rapid Update Cycle

Each weather product may use grid-based values, and their horizontal resolutions are listed in Table 5.3. For the weather products that do not use horizontal resolution, the parameter value is a dash (-).

Table 5.3. Supported Resolution Parameter Values

Value	**Description**	**Product**
`-`	Not used	CWAM
`1km`	1-km horizontal grid	CIWS
`13km`	13-km horizontal grid	RR and RUC
`40km`	40-km horizontal grid	RR and RUC

Certain weather products store data in GRIB format. Currently, there are two editions, and their parameter values are listed in Table 5.4. For weather products that do not use GRIB, the parameter value is a dash (`-`).

Table 5.4. Supported Edition Parameter Values

Value	**Description**	**Product**
-	Not used	CIWS, CWAM
1	Edition 1	RR, and RUC
2	Edition 2	RR and RUC

A weather product may provide more than one weather type. Their supported parameter values are listed in Table 5.5. Additional types can be supported by introducing a new subpackage in the dataset package and adding the following subclasses to the new package:

- Field: Stores supported grid names defined in the data files.
- Specific field classes that represent data fields.

Table 5.5. Supported Type Parameter Values

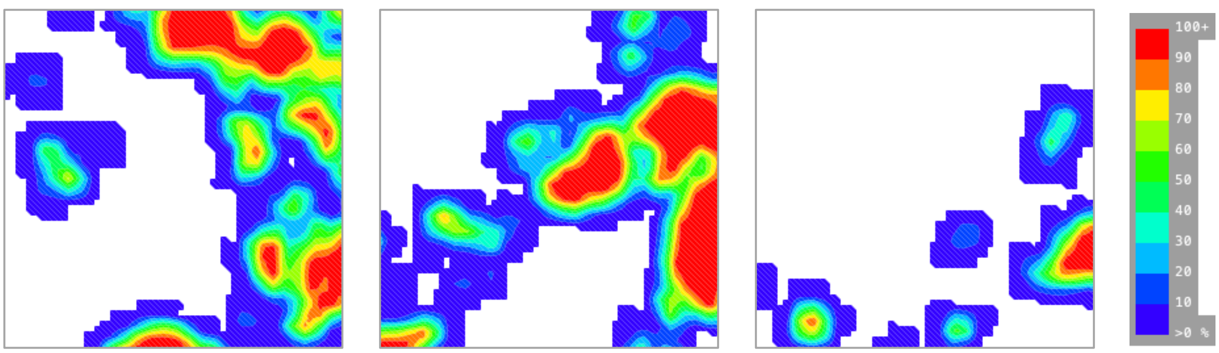
Value	**Description**	**Product**
`cloud-cover`	Cloud coverage, in percentage, based on altitude value, in feet: • [0,6500) = low cloud layer • [6500,20000) = middle cloud layer • [20000,40000) = high cloud layer • [40000, +inf) = entire atmosphere.	RR and RUC
`dp`	Deviation probability, in percentage	CWAM
`echo-top`	Echo top, in feet, of the entire atmosphere.	CIWS
`wind`	Wind magnitude at isobaric surface: • u- and v-component values, in meters-per-second, for data • calculated value, in knots, for tile	RR and RUC
`vil`	Vertically integrated liquid water, in kg/m^2, of the entire atmosphere.	CIWS

Table 5.6 lists the values of eight functions currently supported by the TestBed Weather Image Server. Though the functions, except `first` and `last`, remove all the `NaN` weather data values, they return `NaN` whenever there are no weather data values.

Table 5.6. Supported Function Parameter Values

Value	Description
`avg`	Average of the non-`NaN`[3] weather data values.
`count`	Number of the weather data values that are not `NaN`.
`first`	Weather data value at the first specified time.
`last`	Weather data value at the last specified time.
`max`	Maximum of the non-`NaN` weather data values. This is the default function whenever the apply parameter value is not specified.
`med`	Median of the non-`NaN` weather data values. Non-`NaN` values are sorted in ascending order. The middle value is returned. In case there are two middle values, their average is returned.
`min`	Minimum of the non-`NaN` weather data values.
`sum`	Total of the non-`NaN` weather data values.

Figure 5.1 shows the tile images based on the cloud coverage values at midnight (00Z) of three consecutive days: (a) 7 July 2019, (b) 8 July 2019, and (c) 9 July 2019. The coverage percentages are indicated by a blue-to-red colormap legend shown in (d), where blue region represents low coverage in `(0,10]`% and red region represents high coverage in `[90,100+)`%. White regions indicate no coverage data, i.e., 0%.

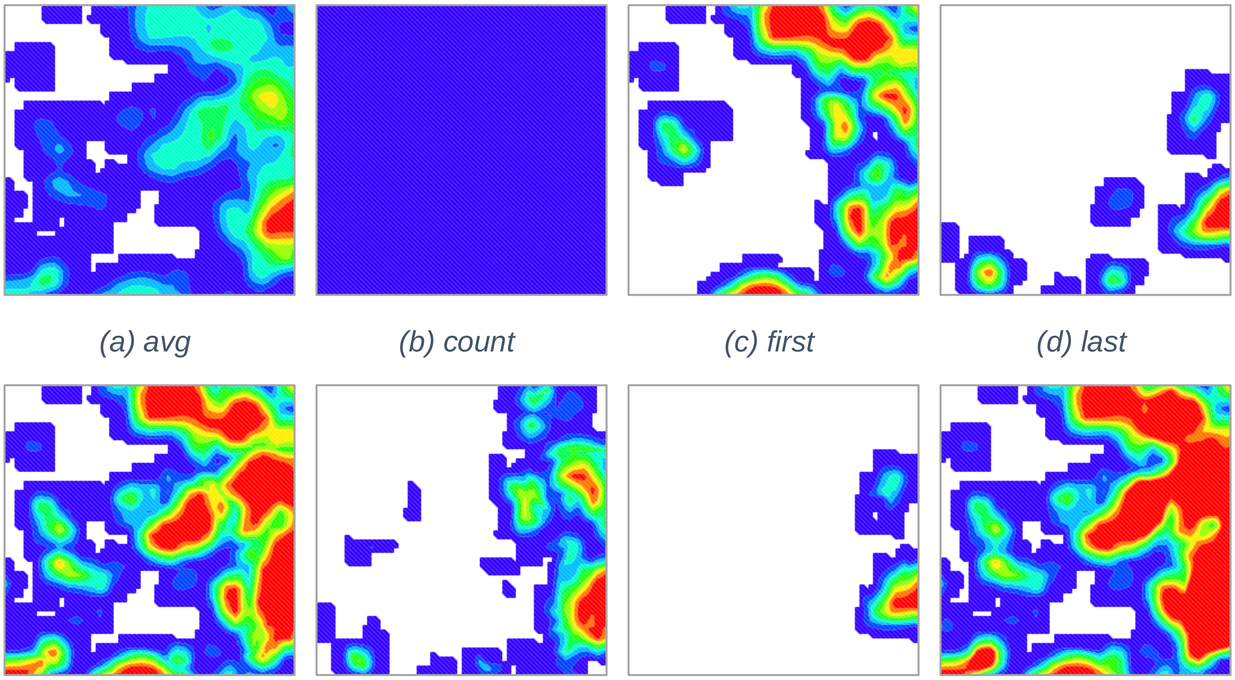


(a) 2019-07-07T00Z *(b) 2019-07-08T00Z* *(c) 2019-07-09T00Z* *(d)*

Figure 5.1. Cloud Coverage Tile Images

Figure 5.2 shows the eight tile images when individual value functions are applied to the coverage data on these three days. The count function returns a dark blue tile because each pixel has a value of three (3) which is in `(0,10]`%.

[3] `NaN` represents a Not-a-Number value of types float or double.



(a) avg (b) count (c) first (d) last

(e) max (f) med (g) min (h) sum

Figure 5.2. Cloud Coverage Tile Images with Applied Functions

5.2. Tile Request URLs

The map tile images representing the interpolated numeric values from the weather data files can be obtained by accessing the tile request URLs, which have the following parameterized format:

```
http://<host>:<port>/<dataset>/<resolution>/<edition>/<type>/<time>/
<forecast>/<altFt>/tile/<zoom>/<tileY>/<tileX>[?apply=<func>]
```

where the parameter fields follow the definitions defined in Table 4.1 and Table 5.1. The difference between a data request URL and a tile request URL is the query location. For data values, the location is specified by data/<latDeg>/<lonDeg>; for tile images, the location is specified by tile/<zoom>/<tileY>/<tileX>. In addition, the image server supports the colormap legend image by accessing the following parameterized format:

```
http://<host>:<port>/legend/<dataset>/<type>
```

where the parameters follow the definitions defined in Table 5.1. An example legend image, see Figure 5.1(d), for the cloud-coverage of the rapid refresh dataset is listed below.

```
http://localhost:7890/legend/rr/cloud-cover
```

5.3. Traffic Viewer

In Build 2.3 of TestBed, a weather tile layer is added to the Traffic Viewer, a two-dimensional visualization tool for showing the vehicles, airspace boundaries, and map tile images. The section of the properties defining the CWAM weather provider in the configuration file named TrafficViewer.properties is listed in Listing 5.3. The property keys and their values are in black and blue, respectively.

Listing 5.3. CWAM Weather Provider Properties

```
weather_provider.NASA_Sherlock_CWAM.name = NASA Sherlock Weather:
  Convective Weather Avoidance Model
weather_provider.NASA_Sherlock_CWAM.attribution = National Oceanic and
  Atmospheric Administration, National Weather Service, and National
  Centers for Environmental Prediction
weather_provider.NASA_Sherlock_CWAM.source = NOAA, NWS, NCEP
weather_provider.NASA_Sherlock_CWAM.maxSouthEastTileUrl =
  http://localhost:7890/cwam/-/-/dp/${dateTime:$SimDateTime}/
  ${forecastMinute:0:0}/${altitude:25000}/tile/23/8374868/8388607
weather_provider.NASA_Sherlock_CWAM.legendUrl =
  http://localhost:7890/legend/cwam/dp
weather_provider.NASA_Sherlock_CWAM.maxZoomLevel = 19
weather_provider.NASA_Sherlock_CWAM.loaderTimeout = 0-0-0T0:1:0
weather_provider.NASA_Sherlock_CWAM.retentionPeriod = 0-0-0T0:1:0
weather_provider.NASA_Sherlock_CWAM.fileCache = false
weather_provider.NASA_Sherlock_CWAM.enabled = false
```

The “maximum south-east tile URL” property supports dynamic values called macros. Three special values for the macro `${dateTime}` are listed in Table 5.7. The special values ensure that weather tile images on specific days or times can be continuously updated during a simulation run.

Table 5.7. Special Date Time Values

Value	**Description**
`$SimDateTime`	Current simulation date and time. Format: `yyyy-MM-dd'T'HH:mm:ss'Z'`
`$SimDate`	Current simulation date. Format: `yyyy-MM-dd`
`$SimTime`	Current simulation time. Format: `HH:mm:ss'Z'`

To demonstrate the apply function “max” (see Table 5.6) that calculates the maximum weather values from two historical dates, Figure 5.3 shows the Traffic Viewer running on 1 October 2021 at 21:37Z. The simulation time is indicated on bottom-left corner. The weather tile layer displays the CWAM polygons, at 25,000 ft, from the data files covering 6 July 2019 at 20:30Z. The settings can be configured by following six steps:

1. Click on the Weather Tile Layer toolbar button.
2. Select the menu item “Edit Macro Values…”
3. Locate the section titled “NASA Sherlock Weather: Convective Weather Avoidance Model” in the dialog.
4. Specify the value “`2019-07-06T20:30Z`” in the “date time” field.
5. Specify the value “`0:0`” in the “forecast minute” field.
6. Specify the value “`25000`” in the “altitude” field.

Notice that in the bottom portion of the figure, there are few weather conditions in the Albuquerque Air Route Traffic Control Center (ZAB, circled in light blue) and some severe weather conditions in the Fort Worth Air Route Traffic Control Center (ZFW, circled in orange).

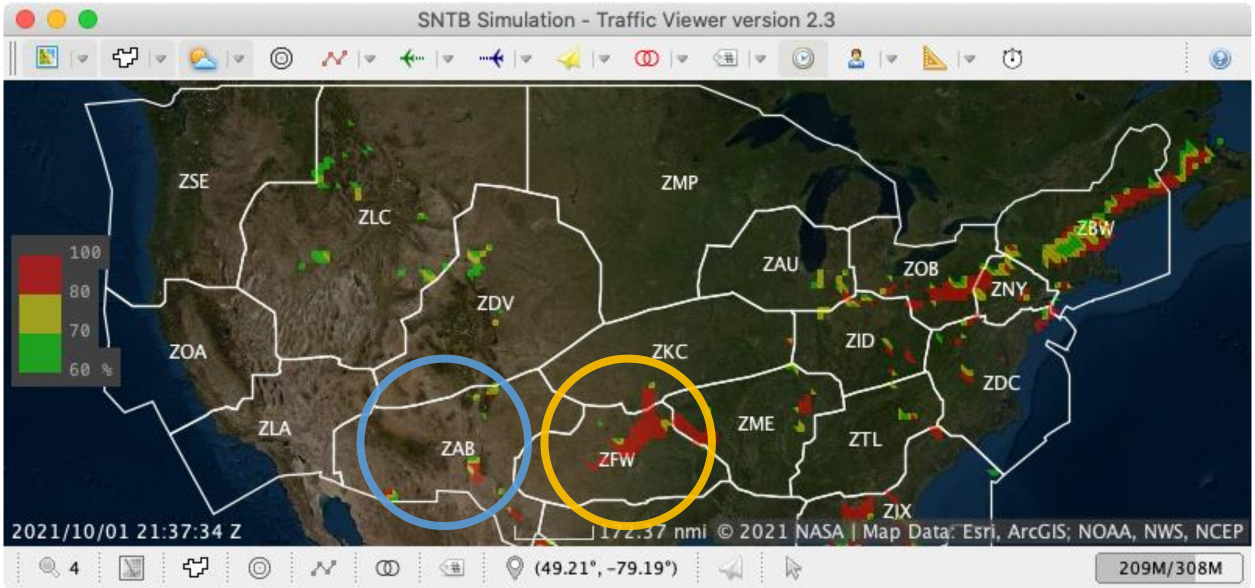


Figure 5.3. CWAM Polygons on 2019-07-06 at 20:30Z

Figure 5.4 shows the CWAM polygons from the data files covering another day, 1 September 2019, at the same time (20:30Z). The value of the “date time” field is specified as “`2019-09-01T20:30Z`” in the Edit Macro Values dialog. Notice that there are some severe weather conditions in ZAB and few weather conditions in ZFW.

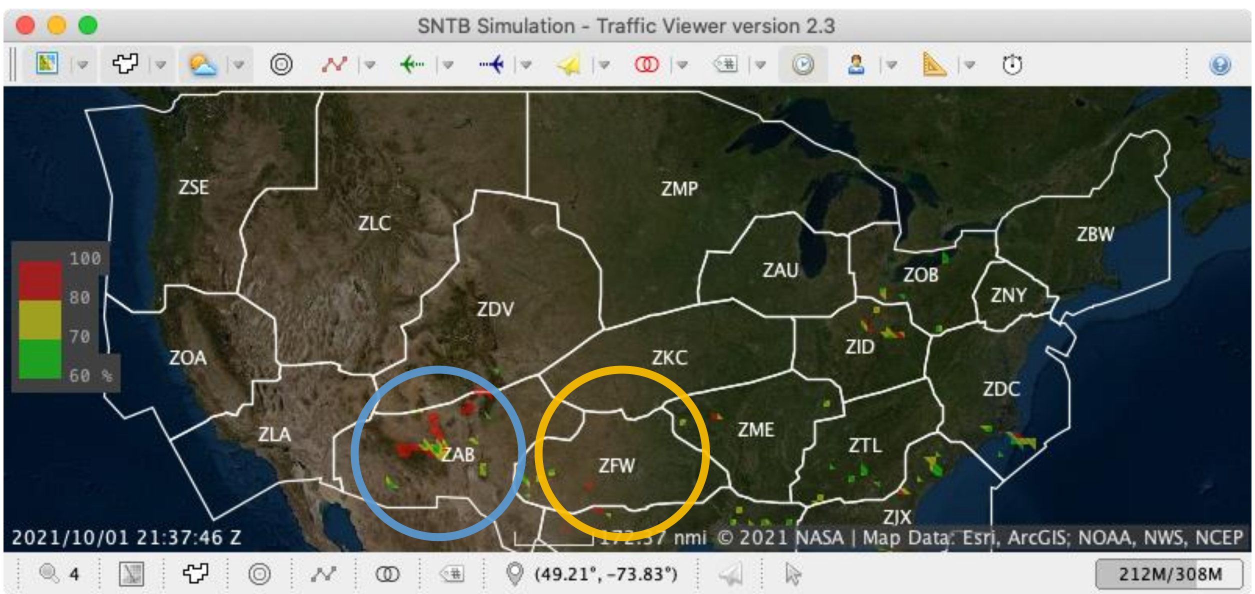


Figure 5.4. CWAM Polygons on 2019-09-01 at 20:30Z

Figure 5.5 shows the combined CWAM polygons from the two days, 6 July 2019 and 1 September 2019, at 20:30Z based on the maximum (`max`) value function. The value of the “date time” field is specified as “`2019-07-06T20:30Z,2019-09-01T20:30Z`” in the Edit Macro Values dialog. Notice that severe weather conditions are shown in both ZAB and ZFW centers. This demonstrates the new capability to allow users to input multiple days of the CWAM data, and polygons aggregated from a function (the maximum function in this example) can be displayed on the Traffic Viewer.

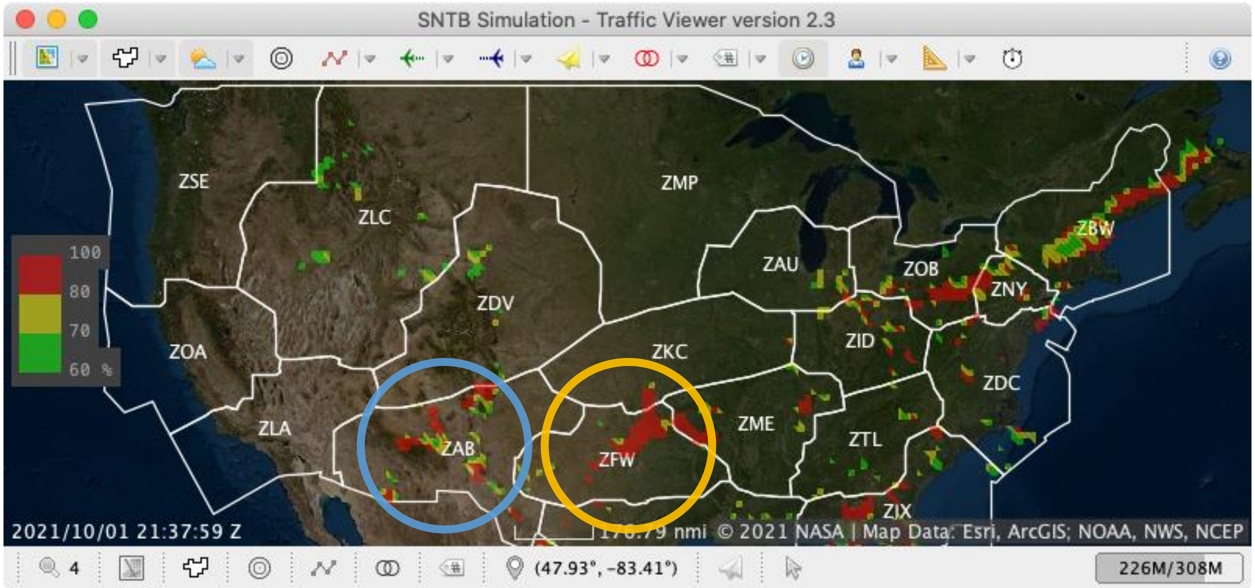


Figure 5.5. Combined CWAM Polygons from 2019-07-06 and 2019-09-01 at 20:30Z

6. Concluding Remarks

This document presents the four weather data products, CIWS, CWAM, RR, and RUC, provided by NASA's Sherlock ATM Data Warehouse for supporting research that can also be accessed via the TestBed weather data service. The TestBed provides standard ways for querying weather data using Java APIs, web data access, and map tile image access. Grid-based weather data values are linearly interpolated to support smooth integration functions. The tile image rendering process uses the triangular meshes obtained by the Marching Squares contouring algorithm. Analysis shows that using four-pixel grids will yield a balance of high image quality and fast image rendering. Finally, TestBed's Traffic Viewer is presented with a newly added weather tile layer, to provide a capability for users to visualize the weather values obtained from a single day or aggregated from multiple days.

7. References

1. The National Imagery and Mapping Agency (NIMA), "Department of Defense World Geodetic System 1984, Its Definition and Relationships with Local Geodetic Systems," NIMA Technical Report TR8350.2, 3 January 2000.
2. Evans, J.E., Weber, M.E., and Moser, W.R., "Integrating Advanced Weather Forecast Technologies into Air Traffic Management Decision Support," Lincoln Laboratory Journal, vol. 16, no. 1, November 1, 2006, pp. 81-96.
3. Krozel, J., "Survey of Weather Impact Models used in Air Traffic Management," 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, 13-15 September 2010, Fort Worth, Texas.
4. Lauderdale, T.A. and Erzberger, H., "Automated Separation Assurance with Weather and Uncertainty," Electronic Navigation Research Institute International Workshop on ATM/CNS, February 19-21, 2013, Tokyo, Japan.
5. Daniels, T., "Using 3-D Numerical Weather Data in Piloted Simulations," NASA/TM-2016-219216, NASA Technical Memorandum, July 2016.
6. Wang, Y., "Weather Impact on Airport Arrival Meter Fix Throughput," 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), St. Petersburg, FL, 17-21 September 2017.

7. Lewis, T.A., Burke, K.A., Underwood, M.C., and Wind, D.J., "Weather Design Considerations for the TASAR Traffic Aware Planner," AIAA Aviation 2019 Forum, Dallas, TX, 17-21 June 2019.
8. Robinson, J.E., Lee, A., and Lai, C.F., "Development of a High-Fidelity Simulation Environment for Shadow-Mode Assessments of Air Traffic Concepts," Royal Aeronautical Society: Modeling and Simulation in Air Traffic Management Conference, London, UK, 14-15 November 2017.
9. "Uniform Resource Locators (URL)" (Online), https://datatracker.ietf.org/doc/html/rfc1738. Internet Engineering Task Force. Retrieved 2021/09/30.
10. Lai, C. F., "Air Traffic Management TestBed Traffic Viewer: Developer's Guide," NASA/TM-2020-220511, NASA Technical Memorandum, April 16, 2020.
11. Eshow, M. M., Lui, M., and Ranjan, S., "Architecture and Capabilities of a Data Warehouse for ATM Research," 2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC), Colorado Springs, CO, 5-9 October 2014.
12. Klingle-Wilson, D. and Evans, J., "Description of the Corridor Integrated Weather System (CIWS) Weather Products," Project Report ATC-317, Lincoln Laboratory, Massachusetts Institute of Technology, August 1, 2005.
13. DeLaura, R., Robinson, M., Pawlak, J., and Evans J., "Modeling Convective Weather Avoidance in Enroute Airspace," 13th Conf. on Aviation, Range and Aerospace Meteorology, ARAM, 20-24 January 2008.
14. "Rapid Refresh (RAP)" (Online), https://rapidrefresh.noaa.gov/. Global Systems Laboratory, National Oceanic & Atmospheric Administration. Retrieved 2021/09/29.
15. "Rapid Update Cycle (RUC)" (Online), https://ruc.noaa.gov/ruc/. Global Systems Laboratory, National Oceanic & Atmospheric Administration. Retrieved 2021/10/05.
16. "Unidata | NetCDF" (Online), https://www.unidata.ucar.edu/software/netcdf/. University Corporation for Atmospheric Research Community Programs. Retrieved 2021/09/30.
17. "GMOS GRIB – MDL – Virtual Lab" (Online), https://vlab.noaa.gov/web/mdl/gmos-grib. NOAA Meteorological Development Laboratory. Retrieved 2021/09/30.
18. "The HDF Group - ensuring long-term access and usability of HDF data and supporting users of HDF technologies" (Online), https://www.hdfgroup.org/. The HDF Group. Retrieved 2021/09/30.
19. "THREDDS Data Server" (Online), https://www.unidata.ucar.edu/software/tds/current/. University Corporation for Atmospheric Research Community Programs. Retrieved 2021/09/30.
20. "Java SE | Oracle Technology Network | Oracle" (Online), https://www.oracle.com/java/technologies/java-se-glance.html. Oracle. Retrieved 2021/09/30.
21. Gosling, J., Joy, B., Steele, G., Bracha G., Buckley, A., and Smith, D., "The Java Language Specification—Java SE 11 Edition," https://docs.oracle.com/javase/specs/jls/se11/html/index.html. Oracle. Retrieved 2021/09/30.
22. "Hypertext Transfer Protocol Version 2 (HTTP/2)" (Online), https://datatracker.ietf.org/doc/html/rfc7540. Internet Engineering Task Force. Retrieved 2021/09/30.
23. "Marching squares – Wikipedia" (Online), https://en.wikipedia.org/wiki/Marching_squares. Wikipedia. Retrieved 2021/09/30.

Figures 5.3-5.5 on pages 20-21:
Source: https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer, Used with permission according to ESRI terms of service: https://www.esri.com/en-us/legal/copyright-trademarks.