5.7   DERIVATION OF MULTIPLE FORECAST FORMATS FROM A GRIDDED
FORECAST DATABASE

Todd C. Dankers *
National Weather Service, Boulder, Colorado

Jeffrey Manion
National Weather Service, Central Region Headquarters, Kansas City, Missouri

1. INTRODUCTION

The Gridded Forecast Editor (GFESuite), developed by Forecast Systems Laboratory (FSL), contains flexible and powerful tools for creating forecast products from a gridded database. These product formats can range from tables and narrative text, to graphical imagery. After a gridded database has been created, numerous GFESuite formatting utilities can be employed to derive forecast products in the various formats. Forecast products can contain general, marine and fire weather information by sampling the array of meteorological parameters available in GFESuite. Since the forecast products are derived from a single database, a greater degree of spatial and temporal consistency within a Weather Forecast Office's (WFO) county warning area can be achieved. The flexibility of the formatter utilities allows local GFESuite administrators to easily implement changes in product format to satisfy the requests of product users. Components of the GFESuite formatting utilities will be described.

Forecast data can be sampled at temporal resolutions as short as one hour and for areas as small as one square kilometer. Spatial and temporal resolution of products is defined by the local configuration of GFESuite.

GFESuite also generates graphical products in Portable Network Graphics (PNG) format for display on internet web pages. Images can be formatted to display multiple parameters and can be labeled with selected sample points. Configuring the format of graphic products is also completely controlled by the local GFESuite administrator.

2. GENERATING ALPHANUMERIC PRODUCTS

Alphanumeric product formatters in GFESuite use the Python programming language

Corresponding author address: Todd Dankers, National Weather Service, 325 Broadway, Boulder, CO 80305-3328;
e-mail: todd.dankers@noaa.gov

(www.python.org) to accomplish their task. This scripting language does not require compiling, which allows a developer to rapidly test changes in Python code for effectiveness. The majority of alphanumeric product generation is handled by a pair of Python utilities; AnalysisMethods.py and TextRulesMethods.py.

AnalysisMethods.py is a file with instructions for the variety of ways that the digital information stored in each parameter of the database can be analyzed and reported to TextRulesMethods.py. Most data in the GFESuite database is stored as a collection of numbers, like a temperature field. This data is referred to as scalar data, and can be analyzed and reported as a single value, as a range of values at a certain time, or as a range of values over a period of time. Advanced statistical information is also available, like the skewness and standard deviation of the data. For vector-type data, a similar variety of analysis methods is available. Special analysis methods to handle the third data type, the Weather parameter, have also been developed. In the case of the Weather parameter, the number of grid points covered by unique weather types is tallied, then AnalysisMethods.py can either report the distribution of each of these to TextRulesMethods.py, or just report the predominant weather occurrences.

By using different analysis methods on a dataset, formatting of products can be tailored to the different needs of product users. For example, one product may need temperatures reported to the nearest degree, while another product uses a range of 10 degrees. Or, a wind speed forecast may need to be reported to the nearest single mph or as a 5 mph range of values. Applying separate analysis methods to the same database allows forecast products to have a different look, but the same underlying information.

After the data has been analyzed according to instructions in AnalysisMethods.py, it is passed to TextRulesMethods.py where the particular form of the forecast product is generated. Text rules are instructions for how to handle the data.

Rules can generate phrases by inserting the necessary value, or values, into a phrase template. For example, the snippet of Python code: *"High temperatures " + minValue + " to " + maxValue +"."* would create a phrase that looks much like a sentence for high temperatures in a typical National Weather Service (NWS) zone forecast. Text rules can be very simple, or they can be very complex, with programming logic to handle the variety of situations that might be reported by AnalysisMethods.py. By matching the data reported by AnalysisMethods.py and the needs of the text rule, there is virtually no limit to what a formatter can produce.

The two files AnalysisMethods.py and TextRulesMethods.py contain the entire collection of instructions for deriving alphanumeric products with GFESuite. These files are intended to be identical at each site. Additional analysis methods and text rules can be written by individual sites to handle situations unique to a particular office. These methods and rules can be placed in a separate file that is accessed by the other product generation utilities. This allows the strength of the formatting utilities to extend to the local office level.

Creating the alphanumeric products requires one, or more additional files which list the specific analysis methods and text rules to employ. These files are product definition files. They are created through the use of a graphical user interface (GUI) in GFESuite. Product definition files are designed as one of three types: table, phrase or smart. Table definition files have the simplest form due to the terse nature of information reported in a tabular product. Output from phrase definition files is generally combined with instructions in a smart definition file to produce the more complex, narrative text products.

2.1 *Forecast Area Information*

The information for locations to be covered by a product is stored in GFESuite as Edit Areas. Edit areas are stored collections of grid points that represent a forecast location, with sizes ranging from a single grid point up to a WFO's entire GFESuite grid domain. Edit areas (Figure 1) can be generated for public zone boundaries, river basins, individual counties or entire county warning areas, using information available in map shape files available in AWIPS. Edit areas can also be hand drawn by a GFESuite user to represent the grid points around an airport, an observation location, or a town or city. Once an edit area has been defined and saved into the GFESuite through use of the Edit Area Query GUI, it is available for use by product generating utilities.
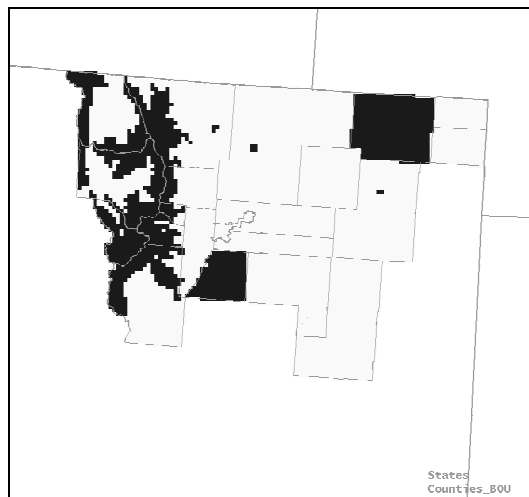

Fig. 1. Selected edit areas within WFO Boulder's county warning area. Edit areas can be defined based on zone, county or town boundaries.

2.2 *Creating Tabular Products*

The simplest type of alphanumeric product to configure is a table. Tabular products can have two basic forms. A product can be a forecast for an edit area, with a range of times and collection of weather parameters, or a product can be for a certain time with a collection of edit areas and various weather parameters. Within a table, columns and rows of information may contain time, location or weather parameter. Tables 1 and 2 are examples of the formats that can be used for tabular products. The appearance of the table; which edit areas to include, time ranges to be used, which weather parameters to use and temporal resolution, if applicable, are all controlled by instructions in the product definition file. Table 3 shows some of the Python coding used to format a product like the one shown in Table 2. Each of these product definition features can be configured for the forecaster to select at run time, or they can all be defined in the product definition file to allow automated generation of products.

```
6 Hour Forecast for Denver from
6 AM MDT Sep 22 TO 1 PM MDT Sep 22.


Time Period       6 AM     9 AM    12 PM


Sky Cover        MOSTLY   PARTLY   SUNNY
                 CLOUDY   CLOUDY
Temperature        54       68       82
Dew Point          32       32       33
Wind (mph)        W  7    NW  7    NW  7
Weather           RAIN
                 SHOWERS
```

Table 1. Sample forecast for a single location over a range of times.

```
Table of Weather for SEP 22, 2001
    12 PM MDT - 1 PM MDT.


Location        Temp(F)  Wind(mph)

Akron              77        N   5
Bennett            79       NW   6
BlackHawk          69       NW   5
Boulder            83       NW   6
CastleRock         79       NW   6
Denver             82       NW   6
```

Table 2.   Sample tabular product with several point forecasts for a single time.

```
## Table Layout
# A table is a combination of three variables:
#   edit areas, weather elements, and time
#   periods.
# One of these variables is held constant,
#   one is assigned to rows and the other to
#   columns.

    "constantVariable": "TimePeriod",
    "rowVariable": "EditArea",
    "columnVariable": "WeatherElement",

"defaultEditAreas": [
          ("Akron", "Akron"),
          ("Bennett", "Bennett"),
          ("BlackHawk", "Black Hawk"),
          ("Boulder", "Boulder"),
          ("CastleRock", "Castle Rock"),
          ("Denver", "Denver"),
       ],

"runTimeEditAreas" : "no", # if yes, ask user
                           # at run time.
"areaType" : "Location",   # E.g. City, County
                           # Basin, etc.

## Weather Elements
"elementList": [("T","Temp(F)",
                 AnalysisMethods.avg
                 TextRulesMethods.singleValue,
                 "Scalar", 1, None),
                ("Wind","Wind(mph)",
                 AnalysisMethods.vectorRange,
                 TextRulesMethods.avgValue,
                 "Vector", 1, None),
                ],
```

Table 3. Snippets of Python code that control the content and format of the product displayed in Table 2.

2.3 *Creating Narrative Products*

Organizing a narrative text product is a multiple step process involving derivation of the component phrases of the narrative, then combining them into the desired sentence, or collection of sentences.   Piecing together different collections of phrases and sentences allows a product to have variations in detail, depending on the lead-time of the forecast. The first day and a half of the NWS's zone forecast product contains information about sensible weather, sky cover, winds, temperatures, probability of precipitation and snow amounts, if applicable. In contrast, the portion of the zone forecast product for days four through seven contains only information about sensible weather and temperatures. Tables 4 and 5 show how different parameters are assigned to the separate time periods of the forecast product. The current GFESuite formatter for generating NWS public zone products contains four levels of information detail. The highest level of detail applies to the first three periods, a second level for the next two or three periods, a third level handles the third day and following evening of the forecast, and the final level of detail is for the last four days of the product.

In the case of the public zone formatter, at each level of detail, a phrase product definition file is needed. In each of these product definition files, the list of parameters to use, the analysis method to employ, and the text rules to apply at this level is included. At runtime, these product definition files will produce a collection of phrases, one phrase for each parameter that was listed, according to the desired text rule.   A product definition file can be run using forecaster input of desired Edit Area and time range. However, the output from a single product definition file is of little value, on its own. The product gains its utility when output from several product definition files is combined in an orderly manner.

Organizing the output from individual product definition files into a coherent forecast product, requires another file, a product format file. This file is a smart file and contains the information about which Edit Areas to use and what forecast times to apply to each product definition file. The NWS public zone forecast actually contains a different number of periods, depending on the time of day that the forecast is being issued. This difference in the number of periods is controlled by the product format file, using input from the forecaster at runtime.   This file also contains instructions on how to label each time period in the product, for example, "Today", "Tonight", the day of the week, or any upcoming holidays.

Generating a zone product involves the forecaster executing the product format file from GFESuite's "Generate Products" menu. This file executes the designated collection of product definition files, which in turn draw their information from the Edit Areas stored in GFESuite, and output from AnalysisMethods.py and TextRulesMethods.py.

```
"analysisList": [
("MinT", AnalysisMethods.stdDevMinmax),
("MaxT", AnalysisMethods.stdDevMinmax),
##          "MinT", AnalysisMethods.minmax),
##          "MaxT", AnalysisMethods.minmax),
("T", AnalysisMethods.hourlyTemp),
("Sky", AnalysisMethods.scalarMedianRange),
("PoP", AnalysisMethods.filteredMinmax),
("SnowAmt", AnalysisMethods.minmax),
("Wind",AnalysisMethods.vectorMedianRange),
("Wx", AnalysisMethods.dominantWxRange),
("WindChill", AnalysisMethods.minmax),
("HeatIndex", AnalysisMethods.minmax),
],

"textRules":[
    TextRulesMethods.reportTrends,
    TextRulesMethods.wind_summary2,
    TextRulesMethods.sky_phrase,
    TextRulesMethods.weatherRange_phrase,
    TextRulesMethods.severeWeather_phrase,
    TextRulesMethods.snow_phrase,
##      TextRulesMethods.highs_phrase,
##      TextRulesMethods.lows_phrase,
    TextRulesMethods.highsRange_phrase,
    TextRulesMethods.lowsRange_phrase,
    TextRulesMethods.wind_phraseWithGusts,
    TextRulesMethods.popMax_phrase,
    TextRulesMethods.windChill_phrase,
    TextRulesMethods.heatIndex_phrase,
    ],
```

Table 4. Python code snippets listing the weather parameters, analysis methods and text rules that are used to generate the first three periods of a NWS zone forecast.

Figure 3 shows how forecast format information is passed through the GFESuite product generation process. By configuring product definition files to use data specific to public, marine or fire weather customers, and setting product format files to organize coherent products, many types of useful products can be derived from a WFO's digital database. It is also possible to embed tabular product formats within a narrative product. With the development of these formatting utilities in GFESuite, it is now possible for developers and users at the WFO level to design and implement forecast products in new forms. Standardization of national products, and the flexibility to create new products to satisfy local needs can be achieved through the judicious use of the strengths contained in these formatting utilities. Details for setting up product definition and product format files are available in GFESuite's Online Help, in the GFE Training Guide.

3. GENERATING GRAPHICAL PRODUCTS

Products in graphic format are generated by a utility called ifpImage. This utility effectively performs a screen capture of the GFESuite spatial editor in unique configurations that are

```
"analysisList": [
("MinT", AnalysisMethods.firstAvg),
("MaxT", AnalysisMethods.avg),
("T", AnalysisMethods.hourlyTemp),
("Wind",AnalysisMethods.vectorMedian),
("Wx", AnalysisMethods.weather),
],

"textRules":[
  TextRulesMethods.highsRange_phrase,
    TextRulesMethods.lowsRange_phrase,
    TextRulesMethods.weather_phrase,
    TextRulesMethods.wind_phrase,
],
```

Table 5. As in Table 4, except for the extended periods of a NWS zone forecast.

controlled by special GFE configuration files. Each configuration file initializes a GFESuite "session" with a limited number of parameters displayed and with specified map settings and other image controls in place. These displays are then captured to file(s) in Portable Network Graphic (PNG) format, using the times contained in the script that is ran the ifpImage utility. Figure 2 is an example of a graphic product displaying temperature and wind information.

Each graphic may contain one image, overlaid with as many contoured fields as desired. The ifpImage configuration file controls almost all of the details about the appearance of the graphic product. This includes the size of the image, the font size and formatting of any legends, the portion of the GFESuite domain to display, which sample points to plot, the thickness of contour lines and whether they are dashed or solid, which color table to use, and many other variables. Details for ifpImage configuration file options are contained in GFESuite's Online Help documentation.
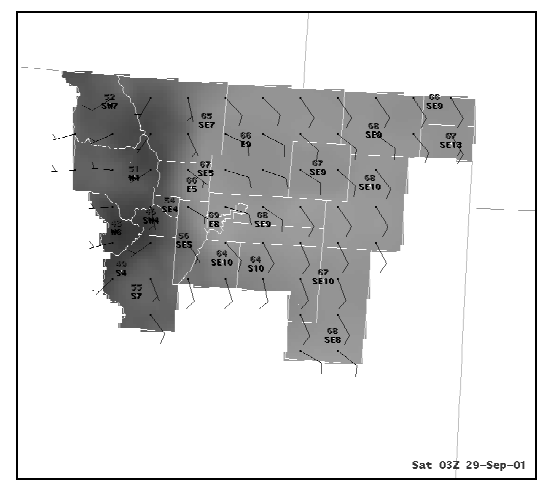

Fig. 2. Graphic image of temperature and wind barbs for WFO Boulder's county warning area.
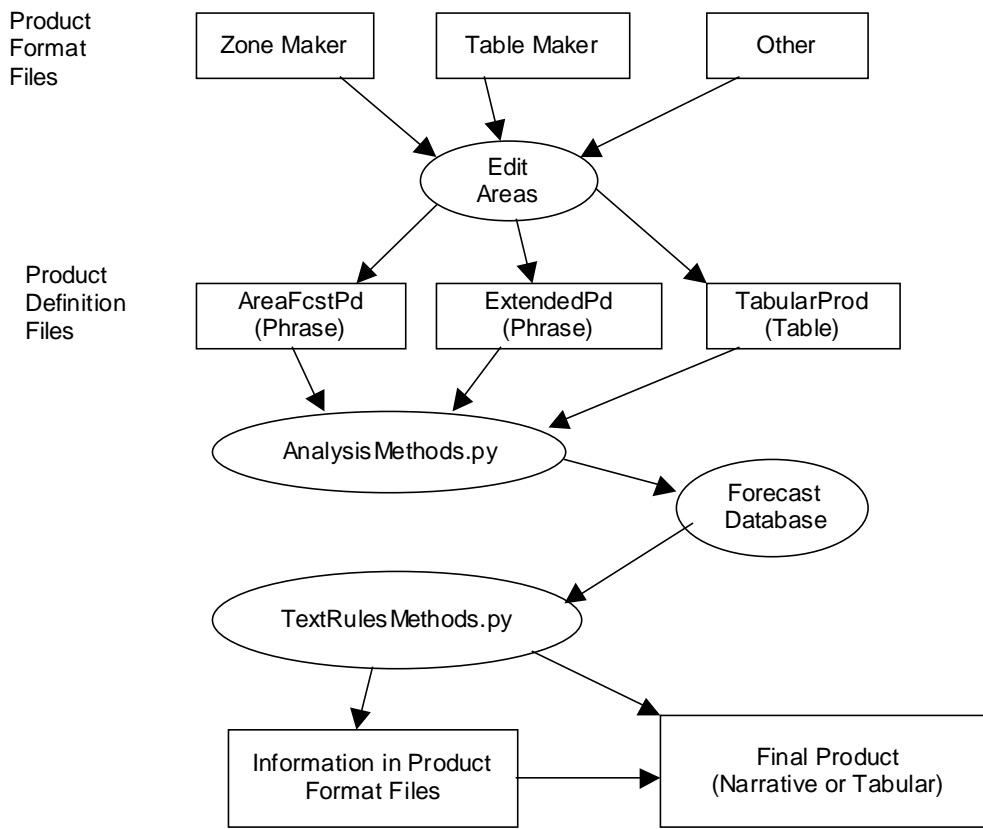
Fig. 3. Schematic of the flow of information during generation of an alphanumeric product.

4. CONCLUSION

GFESuite offers simple methods for developing forecast products directly from the gridded database. An understanding of the relationships between analysis methods, text rules, phrase generation files and product definition files allows a developer at a WFO to develop products that meet the needs of the local customer. Rapid Prototype Project participants have already demonstrated the ability to develop product formats that satisfy local and national product requirements. At the same time, the files necessary for generating nationally standard products can easily be distributed, allowing individual NWS Weather Forecast Offices to quickly begin using the product formatters operationally. Updates to these files can also quickly be distributed to handle bug fixes or changes in product format.

These product generation processes are also independent of the spatial resolution used by GFESuite. This allows offices to run GFESuite at the resolution that suits them best, while having no impact on the alphanumeric and graphic forecast generation process. This independence will also allow offices to easily increase the GFESuite resolution as computer processing speeds increase in the future.

5. REFERENCES

Hansen, T. L., M. Mathewson, M. Romberg, 2000: Forecast Methodology Using the GFESuite. Preprints, 17th International Conference on Interactive Information and Processing System for Meteorology, Oceanography, and Hydrology, Albuquerque, NM, Amer. Meteor. Soc.

Lefebvre, T. J., M. Mathewson, T. Hansen, M. Romberg, 2000: Injecting Meteorology into the GFESuite. Preprints, 17th International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Albuquerque, NM, Amer. Meteor. Soc.

Paxton, C. H., T. Hansen, 2002: GFE Methodology and Extended Capabilities. Preprints, Interactive Symposium on AWIPS, Orlando, FL, Amer. Meteor. Soc.