MeteoAssert: Generation and organization of weather assertions from gridded data

Stephan M. Kerpedjiev
NOAA/ERL/Forecast System Laboratory
325 Broadway
Boulder, Colorado 80303

E-mail: kerpedjiev@fsl.noaa.gov

Abstract

MeteoAssert, a system developed at the Forecast System Laboratory, analyzes gridded data sets and produces descriptions, organized sets of assertions representing the content of weather messages. Each assertion conveys a single weather characteristic with a certain spatial and temporal scope. The assertions in a description are linked by discourse relations that predetermine the structure of the weather message: a natural language text, a piece of graphics, a table, or a mixture of these elements. The descriptions are generated in response to queries representing the information needs of the user. Three models drive the system: territory, time, and parameter. Each model defines the objects in terms by which the descriptions are created. MeteoAssert works as a server to several systems dealing with different applications and preparing various weather displays.

AI topic: *multimodal generation, assertion processing;*
Domain area*: Weather report generation;*
Language/Tool: *IBM PC, Windows 3.1, Microsoft Visual C++;*
Status*: Research prototype in use at the Forecast Systems Laboratory;*
Effort*: One person-year;*
Impact*: The generation of well-organized assertions eliminates the need to manually analyze large amounts of data for generating various weather displays.*

1. Introduction

Every minute a large number of weather data are produced. They refer to either past or future weather conditions and are intended to help people plan their activities so that they will not be adversely affected by the weather. However, often the data are not used appropriately for different reasons: misinterpretation, overloading the weather message with irrelevant information, or missing important elements. To improve the utility of weather information to end users, we need to increase the quality of the weather data, and make these data available to the user in a form that is efficient and supports the user's decision-making process.

The path of any weather information begins with the collection of data using human observations, instruments, radar, and satellite photos. The data continue processing as powerful analysis and prediction models produce gridded weather data sets. The path ends when generated weather reports are interpreted by the particular user or group of users. Although the data collection and processing are based on certain physical laws and are driven by numerical models, the interpretation is a human oriented process, and as such is subjected to information distortion inherent to human information processing.

Weather information interpretation is a process in which humans perceive the information, match it with their plans and goals, and, if necessary, modify them. The perception of the weather message depends on both its content and the way it is presented to the user. For a long time, text has been the predominant mode of weather information communication. A high point in the automatic generation of weather forecast texts is FoG, a system that generates bilingual marine forecasts from graphical depictions [4].

For technical reasons, the use of graphics has been limited. However, convenient tools are being developed for generating graphical objects, a mode that is getting more and more attention. The fundamental work of Bertin [2] treats graphics as a semiological system similar to language. It presents facts using its own means of expression, and, like language, some methods are better than others for expressing certain types of information. This view to graphics allows us to handle both text and graphical messages in a similar manner, at least at the deeper levels of presentation, where we deal with message content rather than message form.

A recent work by Arens and Hovy [1] generalizes the semiological approach by applying it not only to graphics but also to other modes of presentation. Thus, gradually a semiological theory of multimodal presentation systems has begun to emerge. A clear trend toward using AI techniques for generating multimodal presentations is marked by at least two projects: COMET [3] and WIP [8].

Both systems generate instructional texts supported by illustrations. The text and graphics generators are driven by a higher level module that reasons about the impact that each portion of the multimodal document has on the user, and they adjust portions of content and/or format of the document so that the overall effect on the user will be as close to the desired as possible.

Each mode of presentation can successfully accommodate certain types of information, but can be unsuitable for others. Thus, information about dynamic weather conditions is hard to present on a static map, and, conversely, information about static, spatially distributed weather conditions is effectively presented on a map. Sound is a good choice for conveying alerts and warnings to users that do not continuously monitor the weather situation. When the appropriate mode of presentation is used, the interpretation is easier and the efficiency of the weather product is increased.

This paper addresses the problem of generating portions of content for multimodal weather displays from gridded data using AI techniques. Users express their weather information needs in the form of a query, and the system responds by generating a description, an organized set of assertions. Each assertion conveys a single weather characteristic (the value of a given parameter) associated with a region and time period. The assertions are organized into larger chunks of information using certain discourse relations. This organization, as opposed to unrelated assertions, allows immediate casting of the description into a corresponding language or graphics structure, thus ensuring simplicity of the processing and coherence of the final product.

MeteoAssert is a system designed on the principles described in this paper. It was developed at the Forecast Systems Laboratory where it runs providing real-time information to the dissemination workstation. The flow of information in the dissemination system is completely automatic, which is not the case even with FoG; FoG processes gridded data that have been annotated by a human meteorologist.

Previous work [5] deals with a similar problem: generating multimodal displays from data sets. This work showed that the generation of unrelated assertions, which have to be further analyzed in order to be grouped by any discourse relations, is a possible alternative to the current approach, but it poses severe problems with the efficiency and the complexity of planning products from unrelated assertions.

This paper first describes the content and structure of the weather messages. The three models that outline the scope of the information produced by the system are then presented. Next the objects involved in the computation of descriptions and the transformations involved in this process are specified. Finally, two applications of MeteoAssert are outlined.

2. Weather messages: content and structure

No matter which mode of presentation is used, weather messages consist of assertions. Each assertion conveys a single weather characteristic associated with a region and period. The weather characteristic is the value of a given parameter, e.g., maximum temperature, amount of precipitation, or fire danger. A real application usually deals with tens of parameters, numerical or categorical. The regions are spatial objects known to the user. They can be administrative regions such as counties, or physical regions such as mountains. The periods are temporal objects having a name (such as "morning"). The objects that participate in assertions are specified in corresponding models: parameter, territory, and time.

An assertion is presented in language as a clause or sentence, in maps as a pictograph placed in a certain position over the map, in diagrams as a bar, or in graphs as a line segment. The particular choice of means of expression (words, syntactic structures, pictographs, or table structures) is a problem that is solved within the presentation system, and is out of the scope of this paper.

The message is not just a collection of assertions; it is an organized set of assertions. The assertions are organized so that they make a coherent whole, which greatly facilitates the perception and assimilation of the contents by the user. Various relations are used to organize the contents of a message. These relations must have counterparts in at least one of the various presentation systems; otherwise, they cannot fulfill their function of providing the organization that, after being cast into the corresponding presentation system, will ensure efficient perception by the user. I call these relations discourse relations by analogy of the relations used to organize text and conversation.

The following discourse relations are common in weather messages:

- *Parametrical description.* The elements of this relation are assertions or subdescriptions that refer to different but related parameters.
- *Spatial description.* The elements refer to regions linked in a certain way.
- *Temporal description.* The elements refer to periods linked in a certain way.
- *Value description.* The elements refer to different characteristics of a phenomenon specified by a given value of a given parameter.

The relations among parameters, regions, or periods used in the descriptions are specified in the corresponding model.

The multimodal display in Fig. 1 features five modes of presentation: signal lights, a map, a line diagram, a natural language text, and a list. The signal light color encodes the existence of a certain threat. The map conveys the spatial distribution of certain parameters (the categorical values are encoded by means of icons). The line chart represents the development of a critical parameter contributing to the danger indicated by a signal light. The text portion provides summary information and certain intermodal links

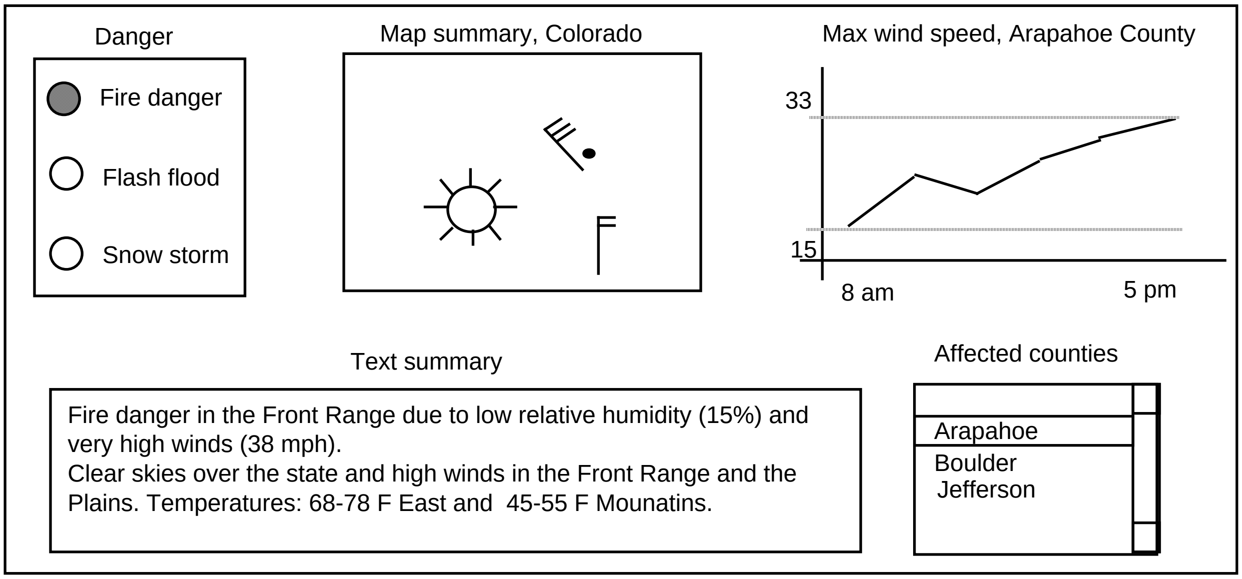

Fig. 1. An example of an assertion-based multimodal display.

(e.g., the link between a threat indicated by a signal light and a critical parameter). The list of regions shows the counties in which the fire danger is in effect. The signal lights are created from parametrical descriptions, the map and the list of regions from spatial descriptions, the line chart from a temporal description, and the text from mixed descriptions.

3. Models

Three models with similar structures drive the process of generation. These models are repositories of knowledge shared among the users and the system about objects involved in weather messages. Each object consists of the following elements: name and type, carrier, and two types of relations: immediate predecessor and coverage. The name and the type of an object are labels used to specify the object itself and the group to which it belongs, respectively. The carrier defines the object. The relation of immediate predecessor allows the computation of the relation of predecessor and successor between any two objects in a given model as explained below. The relation of coverage allows the generation of optimal descriptions.

The carrier has a different meaning and format in the three models. The carrier of a region is its area: a set of grid points. The carrier of a period is a time interval. The carrier of a parameter is its domain (the set of values), the unit, and a sequence of rules that compute the value of the parameter (this third part of the carrier of a parameter is called computation).

The relation of predecessor (superobject) and successor (subobject) is often used to organize weather messages. This relation has different meanings in the three models though. A region A is a superregion of B if the area of A contains the area of B. Likewise, a period A is a superperiod of B if the interval of A contains the interval of B. In the case of parameters, there are parameters of type aggregate that are collections of other parameters representing a certain aspect of weather. Such aggregates are designated as superparameters of the parameters they collect. To compute the relation of precedence between any two objects of a model, we use the relation of the immediate predecessor. An object A is an immediate superobject of B if A is a superobject of B, and there is no other object C that is subobject of A and superobject of B. The relation of the immediate predecessor enables the objects of a given model to be arranged in a lattice.

A coverage of a given object is a set of subobjects that taken together can replace the object. In the case of regions, for example, among the numerous subregions of the state of Colorado, the three subregions of East Colorado, Central Colorado, and West Colorado represent a coverage of Colorado because they are disjoint and the union of their areas is equal to the area of Colorado. Another coverage is North and South Colorado. When a query for optimal description is given, the system analyzes the particular weather data and selects among the different coverages the one that optimizes the description: makes it most precise or most compact.

In summary, we regard the three models as separate knowledge bases consisting of objects. Each object is represented as a frame-like structure with slots for name, type, immediate predecessors, coverages, and carrier, which in the case of a parameter has two subslots: domain and computation. The slots of immediate predecessor and coverage interlink the members of a model.

The region given below covers Boulder County. Its carrier spans from x-coordinate 19 to 24 and from y-coordinate 38 to 41. Boulder County has one immediate

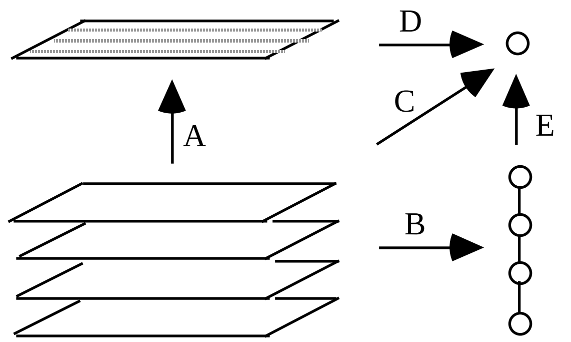

Fig. 2. The basic data carrier objects and the summarizations between them: the cube at the lower left corner consists of four grids and the time series at the lower right corner consists of four values.

superregion and two coverages: East and West Boulder, and North and South Boulder.

```
Name: Boulder;
Type: county;
Carrier: (19,24,38,41);
Immediate superobjects: Colorado;
Coverages: (E_Boulder, W_Boulder),
           (N_Boulder, S_Boulder);
```

4. Objects and transformations

The process of description generation is a sequence of transformations applied to certain objects. In this section, the basic objects and transformations are introduced.

4.1. Objects

There are three basic types of objects: data carriers, spatial and temporal objects, and descriptions. The following data carrier objects are used in MeteoAssert:

- *Grid.* A matrix of values, each one representing the state of a weather variable in a grid point.
- *Cube.* A sequence of grids representing the state of a weather variable in successive moments[1].
- *Time series.* A sequence of values representing the state of a weather variable over a region in successive moments.
- *Value.* Represents the state of a variable over a region and during a period.

The initial data to the system are grids produced by numerical models. Cubes are used to accommodate all the grids of a variable over a period. Time series and values are used to accommodate summarizations of cubes or grids over spatial and/or temporal objects. The data carrier objects are illustrated in Fig. 2.

The following are the spatial and temporal objects used in MeteoAssert:

- *Area*. A set of grid points.
- *Interval*. A set of moments.
- *Cylinder*. A set of areas referred to successive moments.

The area is a shape in a grid, the interval is a segment of the time axis, and the cylinder is a body inside a cube. We distinguish two types of cylinders: a straight cylinder (one created by combining a region and period), and a floating area (a cylinder that consists of those cube points in which the values of a variable satisfy a given condition). In our system, the cylinder is not necessarily a contiguous body.

The descriptions form a separate class of objects. Although they are data carriers, their structure is more complex and subject to specific transformations. They are hierarchical structures of tuples (nodes), each tuple representing either an assertion or a discourse relation. A tuple has seven elements: type, parameter, region, period, cylinder, value, and precision rate. The type specifies if the tuple is an assertion, or, if not, the particular discourse relation. The parameter, region, and period are elements of the parameter, territory, and time models, respectively. The cylinder is either a straight cylinder obtained from the region and the period, or a floating area. The value is used in assertions only and usually summarizes the assertion variable over the assertion cylinder. The precision rate, a number between 0 and 1, specifies to what extent the original data are distorted by the assertion or the overall description.

Queries are yet another type of objects. Their structure is the same as the structure of the descriptions, but each node is regarded as a request to generate a subdescription of a certain type rather than as an assertion or a discourse relation. The parameter, region, and period slots of each query tuple specify the weather variable and the spatial and temporal scope of the query.

4.2. Data transformations and methods

We use three transformations: convergence, merge, and summarization, each of which can be applied to some or all of the data objects defined above. The transformations are further specified by methods that explicitly define how the computation should be performed.

The *convergence* transforms a given object into an object of the same type; e.g., a cube into a cube, a value into a value. The method specifies how each particular value of the source object is converted into the corresponding value of the target object. The convergence is used to categorize numerical parameters, or to transform the data items of an object from one scale to another. A typical example of convergence is the computation of the categorical variable of wind force from the numerical variable of wind speed using the Beaufort scale.

The *merge* maps two or more objects of the same type into an object of the same type. The method specifies how the corresponding data items of the source objects are combined into a single data item of the target object. A

[1] Point and moment in this paper refer to an element of a discrete space: grid or discrete time axis.

typical example of merging is the transformation of the source grids of u-wind and v-wind into the target grid of wind speed or wind direction. Another typical example of merging is the computation of derived grids such as fire danger and flood danger which depend on more than one basic variable.

The *summarization* maps a composite object (cube, grid, or time series) into an object of a lower dimension, such as a cube into a grid. In addition to the source and target object, we also need a spatial and/or temporal object to specify the scope of the summarization. Thus the summarization of a grid over an area is a value (transition D in Fig. 2). The method specifies the rule of computing the value from the data items of the grid that belong to the area; e.g., the rule could be "compute the mean value of the data items." The summarization of a cube over an area is a time series (transition B), where each data item of the time series is obtained by summarizing the corresponding grid over the area. The summarization of a cube over a cylinder is a value (transition C), the summarization of a time series over an interval is a value (transition E), and the summarization of a cube over an interval is a grid (transition A). Some typical methods of summarization are mean value, accumulation, maximum and minimum value, predominant category, and trend.

4.3. Spatial and temporal object transformations

The following transformations of spatial and temporal objects are defined and used in the generation of descriptions:

- Two types of cylinder creation: composition of a straight cylinder from an area and duration, and creation of a floating area within a cube.
- Three types of cylinder decomposition into subcylinders: temporal slicing of a cylinder by subperiods, regional slicing of a cylinder by subregions, clustering of a cylinder into contiguous subcylinders.
- Summarization of spatial and temporal objects. The result of this transformation is a value or time series. The following methods are often used: area, length, volume, x-center of mass, y-center of mass, radius. For example, the volume of a cylinder is the number of cube points that belong to the cylinder.

4.4. Assertion and description generation

Descriptions are generated from queries. Each query specifies the general discourse structure of the description and its spatial, temporal, and parametrical scope. It has the format of a description but its structure may be incomplete, and the values of the assertions are missing. As a result of the generation, the system completes the structure of the query, fills in the slots of the assertion values, and computes the precision rates of all nodes. Each tuple in the query is regarded as a subquery processed according to a method specific to its type. The process of generation is recursive, starts from the root of the query, and proceeds in a depth-first manner. The leaves of the query are requests for assertions.

The processing of a request for assertion consists of summarizing data carriers or spatial and temporal objects into the value of the node. The sequence of transformations is defined in the computation slot of the assertion parameter, as specified in the parameter model.

The processing of an internal query node results in the generation of a description tuple representing a certain discourse relation. The successors of the description node are produced from a pattern given as a successor of the corresponding query node. This pattern may be multiplied as many times as the members of the corresponding discourse relation.

The following query types can be used:

- *Parametrical description.* Generates a parameter description node, i.e., one combining subdescriptions that refer to parameters that are related in a certain way. The relation could be all immediate subparameters of the current parameter, or the subparameters of a given type.
- *Regional description.* Generates a regional description node, the immediate successors of which refer to regions that are in a given relation with the current region. Again the relations of super and subregion can be used as defined in the territory model.
- *Temporal description.* Generates a temporal description node. Its successors refer to periods that are in a given relation with the current period. The relations of a super and subperiod can be used as defined in the time model.
- *Value description.* Generates a value description node and a floating area that is assigned to the tuple cylinder as well as to the cylinders of the immediate successors of the current node.
- *Location description.* Generates a regional description node. This type of query is applied when the cylinder is a floating area. The successors of such a node refer to the same parameter and period as well as to regions that are in a certain relation with the floating area: e.g., the regions cover a certain fraction of the floating area, or the floating area covers certain fractions of the regions.
- *Optimal description.* Generates one or more tuples of regional or temporal descriptions. Decides which of the two discourse relations will be chosen, and which regions or periods will participate in the relations. All these choices are made by the system trying to optimize a criterion given in the query node. The criterion could be either maximum precision rate given an upper bound of the successors to be generated, or minimum number of successors given a lower bound of the precision rate. The meanings of these two criteria are to generate the most precise description from those that do not exceed a given

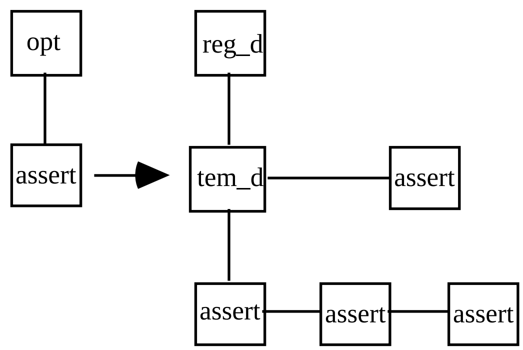

Fig. 3. The structures of the optimal query (to the left) and the description produced from it (to the right).

number of assertions, or to generate the most compact description from those whose precision rates are not less than a given threshold.

Examples

The following examples illustrate the process of description generation. The notation of MeteoAssert was slightly changed to make the queries and descriptions readable. The number preceding each node indicates its rank, i.e., how deep the node is in the hierarchical structure. After the rank, the tuple elements are given in the following order: type, parameter, region, period, value, and precision rate. Some of the last elements of a query node may be void.

The following query is a request for an optimal description of the predominant cloud cover over Boulder county and during the whole day.

```
1 opt(max_pr, succ<5) pc_cloud Boulder whole_d
2 assert
```

The optimization criterion is the maximum precision rate among all descriptions having less than 5 successors. The description generated in response to this query consists of six tuples, of which four are assertions and two are discourse relations (the query and description structures are shown in Fig. 3):

```
1 reg_d void Boulder whole_d void 0.69
2 tem_d void E_Boulder whole_d void 0.72
3 assert pc_cloud E_Boulder morning clear 0.75
3 assert pc_cloud E_Boulder afterno mos_c 0.71
3 assert pc_cloud E_Boulder evening par_c 0.70
2 assert pc_cloud W_Boulder whole_d clear 0.65
```

At the highest level the discourse relation is a regional description by two subregions: East and West Boulder County. The node of East Boulder represents another discourse relation of type temporal description by three subperiods: the morning, afternoon, and evening. All the successors of the temporal description node, as well as the second successor of the regional description node, are assertions. This description can be verbally described as follows:

```
In East Boulder it was sunny in the
morning, mostly cloudy in the afternoon,
and partly cloudy in the evening. In West
Boulder, it was clear throughout the day.
```

In another situation, the system might create another description in response to the same query. If hypothetically the weather was sunny in North Boulder and cloudy in South Boulder throughout the whole day, the system would generate the following description:

```
1 reg_d void Boulder whole_d void 1.00
2 assert pc_cloud N_Boulder whole_d clear 1.00
2 assert pc_cloud S_Boulder whole_d mos_c 1.00
```

The following query is a request for a storm description.

```
1 val_d(>) refl Boulder last_m 35.0
2 assert volume
2 locate(cities)
3 assert max_refl
```

Values of the reflectivity parameter greater than 35 dBz indicate the existence of storm activity. The query for value description generates a floating area over Boulder County during the last moment. The floating area contains those cube points that belong to Boulder County and to the last moment and in which the values of the reflectivity parameter are greater than 35 dBz. The following nodes are queries about certain aspects of the floating area. The first assertion query will generate an assertion about the volume of the floating area. The locate node will search for cities that belong to the floating area and hence are under the storm. For each such city an assertion will be generated giving the maximum reflectivity over that city. A description generated in response to this query is given below:

```
1 val_d refl Boulder last_m 35.0 1.0
2 assert volume void void 5.0 1.0
2 locate void void void void 1.0
3 assert max_refl Longmont void 37.5 1.0
3 assert max_refl Lyons void 48.7 1.0
```

This description shows that the storm is over an area consisting of five grid points and over the cities of Longmont and Lyons where the maximum reflectivity is 37.5 and 48.7 dBz, respectively.

5. Implementation and utilization

MeteoAssert is a server to several application systems having their own presentation modules. It accepts queries from the applications, processes them, and returns descriptions. Some of the queries are triggers that are processed every time new data are received. The descriptions generated in response to triggers are then accessible throughout the whole period until the next portion of data is received. Other queries are generated in the process of exploring the weather situation by the user of the application system. Such queries called real-time queries are passed on to MeteoAssert which processes them and returns the descriptions to the application system. A technology of providing hypermedia access to weather information through assertions is described in [6].

The source data for MeteoAssert are gridded data sets produced by the Local Analysis and Prediction System (LAPS). LAPS produces data every hour on a 10 km resolution grid; the domain of the grid is 61 by 61 over East and Central Colorado as well as small portions of Wyoming, Nebraska, Kansas, Oklahoma, and New Mexico. The system keeps track of the last 24 grids, thus generating descriptions for a whole day.

The system uses three territory models: one consists of all counties and weather zones of Colorado that fall into the scope of the grid, as well as some major cities, and three large state parts: the plains, the foothills, and the mountains. The second territory model contains four river basins in Colorado and their subbasins. The third model includes the roads of Colorado and many road sections.

Currently the system is used to create surveillance and summary products. The surveillance product presents information about hazardous phenomena, such as storms, fog, heavy rain, or dangerous wind-chill, occurring in the last hour somewhere in the state. Descriptions produced from triggers are presented in two modes: graphics and text. In the graphical mode, the occurrence of a certain dangerous phenomenon is indicated by a red light on a panel; the non-existence of the phenomenon is indicated by a green light. The monitor lights are updated every hour as soon as the new data are received and the corresponding triggers are processed. If a hazardous phenomenon has been detected, the user can request a text product that gives information about the counties that are affected by the phenomenon, and the severity of the phenomenon over these counties.

The summary product begins with a map on which the basic parameters (temperature, cloud cover, precipitation, and wind) are presented. This is a highly condensed product that gives only a general idea of the weather situation over the last 24 hours: the range of temperatures, the evidence of winds above a certain threshold, the predominant category of cloud cover, and evidence of any precipitation. This information is spatially distributed through the three basic areas of East and Central Colorado: the plains, the foothills, and the mountains. The assertions are represented on the map by means of icons. The user can request more information about a particular parameter, region and period. As a result, temporal and regional distributions of the selected parameter are generated and presented as a graph showing how the values of this parameter changed over the specified period, and as a text product describing the spatial distribution of the parameter in more detail.

MeteoAssert was written in Microsoft Visual C++ and works on an IBM PC under Windows 3.1. It exchanges queries and descriptions with the application systems through text files sent over the network. Most of the queries are processed within one second, but some complex queries may need several seconds.

7. Conclusion

MeteoAssert can be characterized as a system that converts information from gridded data to assertions to descriptions to multimodal weather messages. The specific features of this approach are as follows:

- The source information comes from gridded data sets providing comprehensive representation of the state of the atmosphere. The data sets are directly obtained from analysis and prediction models without any human intervention.
- The atomic contents portions generated by MeteoAssert are assertions: region and period related pieces of information about user-oriented weather variables.
- The assertions are organized in descriptions, which are coherent chunks of information, by means of discourse relations.
- The descriptions are used to generate weather messages in different modes of presentation rather than just natural language text. Other presentation modes include graphics and tables.
- The process of generation is model-driven: the system can be customized to particular applications through a set of models: domain, time, and territory. The models constitute a distributed knowledge base shared by both MeteoAssert and the presentation systems.

MeteoAssert is part of the FSL Dissemination Project [7] that is being developed to provide high-quality weather data to various groups of specialized users, such as emergency preparedness managers.

Acknowledgment.

I would like to thank Dave Small for the overall support, to Denice Walker for reviewing the paper, and to Nita Fullerton for the careful editing of the paper.

This work was done while the author held a National Research Council Research Associateship at NOAA's Forecast Systems Laboratory.

References

[1] Y. Arens and E. Hovy, "How to Describe What? Towards a Theory of Modality Utilization," In: *Proc. 13th Meeting of Cognitive Science Society*, August 1990, Cambridge, MA.

[2] J. Bertin, *Semiology of Graphics*, University of Wisconsin Press, Madison, 1983.

[3] S. Feiner and K. McKeown, "Automating the Generation of Coordinated Multimedia Explanations," *IEEE Computer*, Vol. 24(10), pp. 33-41, 1991.

[4] E. Goldberg, "FoG: Synthesizing Forecast Text Directly from Weather Maps," In: *Proc. 9th Conf. on Artificial*

Intelligence for Applications, March 1-5, 1993, Orlando, Florida, 156-162.

[5] S. Kerpedjiev, "Automatic Generation of Multimodal Weather Reports from Datasets," In: *Proc. 3rd Int. Conf. on Applied Natural Language Processing,* April 1-3, 1992, Trento, Italy, 48-55.

[6] S. Kerpedjiev, "Model-driven hypermedia access to weather information," In: *Proc. 2nd Conf. on Information and Knowledge Management,* November 1-5, 1993, Arlington, Virginia, in print.

[7] D. Small, "The Dissemination Project," In: *FSL in Review Fiscal Year 1992,* 1993, 5-7.

[8] W. Wahlster et al, "Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation," In: *Proc. 5th Conf. European Chapter of ACL,* April 1991, Berlin, Germany.