2.5 GENERATION OF REAL-TIME NARRATIVE SUMMARIES FOR REAL-TIME WATER LEVELS AND METEOROLOGICAL OBSERVATIONS IN PORTS®

Thomas Bethem[1], Janet Burton[1], Ted Caldwell[2], Mike Evans[1],
Richard Kittredge[2], Benoit Lavoie[2] and Jennifer Werner[1]

[1] NOAA Ocean Service, 1305 East West Highway, Silver Spring, MD 20910
[2] CoGenTex, Inc., 840 Hanshaw Road, Ithaca, NY 14850

1. INTRODUCTION

NOAA's PORTS® (Physical Oceanographic Real-Time System) currently provides historical and real-time observations and predictions of water levels, coastal currents and other meteorological and oceanographic data for a dozen US estuaries. The PORTS web site[3] displays real-time data in graphical and tabular form. The tabular form can require lengthy inspection to identify salient conditions and retrieve the most relevant details for the user, particularly in a large PORTS locale with many sensors. The maritime user community typically includes ship pilots, boaters, fishermen, the military, marine planners, first responders and a range of casual users.

ARNS (Automated Real-time Narrative Summaries) is a prototype system being developed by CoGenTex for NOAA's National Ocean Service (NOS) along the lines proposed by Bethem and Burton (2003). In particular, ARNS will complement the tables and graphics by providing automatically generated natural-language summaries of estuary conditions, in order to give users a more comprehensive overview of the data. The narrative form of the summaries can be more readily understandable to casual users while reducing or eliminating the need for all users to visually inspect multiple graphic products to understand the physical oceanographic behavior of the harbor or estuary. Once the system is fully operational, textual summaries will also enable synthesized speech output for users in "hands and eyes busy" situations.

PORTS data is currently updated every six minutes. This imposes a requirement on the performance of the generator, which should take no more than a few seconds to compose a summary for posting to the site. In fact, when different types of summaries are composed for diverse user groups in the future, there should be no discernible delay in the delivery of any type of summary.

One additional requirement on ARNS is critical. The generator and its knowledge-based resources must be easily maintainable by NOAA staff. Previous rule-based text generators dedicated to relatively complex weather summaries (e.g., Goldberg et al. 1994; Kittredge and Lavoie 1998) have relied on multi-layered linguistic representations, which can require a certain amount of training in computational linguistics to maintain and extend. The challenge in building ARNS has been partly to provide enough linguistic information in the rules for sophisticated fluent text while keeping the system modular, efficient and easy to maintain without extensive training.

In the remainder of this paper we present the architecture of ARNS (Section 2), the Exemplars Framework, used to build the generator (Section 3), the XML schemas for data input and the XML templates for specifying the content and structure of reports (Section 4), current ARNS status (Section 5), and discussion (Section 6).

2. ARNS ARCHITECTURE

The basic architecture of ARNS is shown in Figure 1 below. ARNS consists of two main components:

- a Report Generator, which handles runtime generation of textual summaries, based on input data;
- a Template Editor, which is used to define report templates.

The Report Generator is implemented as a Java component intended to be used within an XML Web Service. It receives observation data and parameter settings from PORTS, and returns generated summaries. The summaries are returned as XML documents, which may or may not make use of specific presentation-oriented markup languages such as HTML. The Report Generator is fully implemented but its integration with PORTS is still ongoing (currently the integration is file-based, using data files generated offline). PORTS will be responsible for the

[3] http://co-ops.nos.noaa.gov/d_ports.html

presentation of the textual summaries, for example as part of a web page.

The Template Editor is implemented as an XML Web Service with a web-based user interface, which lets developers edit and store reusable collections of parameter settings that can be passed to the Report Generator to generate various types of textual summaries. These parameter settings allow developers to specify a wide range of different report types, which can be modified and generated on demand without requiring any reconfiguration or interruptions of service in the Report Generator. The Template Editor is also fully implemented.

The formats for input data to these two modules are discussed in more detail in the following sections.

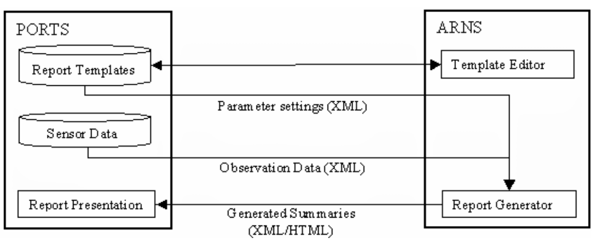

Figure 1. ARNS Architecture

3. EXEMPLARS FRAMEWORK

The ARNS text generator is built using Exemplars, CoGenTex's proprietary framework for surface-oriented text planning[4]. Exemplars provides a hybrid Java/XML source language for defining text planning rules as template-like objects, but with recursivity and an object-oriented rule selection mechanism based on specialization. By defining first general rules, then more specialized rules based on specific data conditions, the Exemplars developer can create generators that cover a wide range of output phrases, but with a much more manageable ruleset than would be possible using traditional templates and simple "if-then" logic. The framework also has modules to handle capitalization, punctuation, spacing, and morphology, which can add great complexity to a simple template-based generator.

Another Exemplars component that makes a surface-based approach more feasible is the ability to define revision rules, which operate on the first-pass output of the generator, typically for purposes of "text smoothing" – for example, introducing sentence connectors such as *and*, *but*, or *also* where appropriate, depending on the structure of the sentences generated on the first pass. As with punctuation, spacing and morphology, postponing certain sentence-planning decisions involving these connectors to a separate revisions stage usually makes the base ruleset more manageable. There is a tradeoff, in that correct operation of revision rules requires a certain amount of syntactic and rhetorical structure to be represented by markup inserted in the first pass. The usual approach is to use "just enough" markup to drive the required rules for a given application; this makes the amount of markup a somewhat ad-hoc decision, but the rules and markup techniques tend to be reusable for different applications with only small modifications.

In order to provide NOAA with the flexibility to customize the generated summaries for different locations or audiences, the generator is instrumented with various configuration parameters, settings for which can be edited in "report templates". These parameters include basic options for including various sensor types (such as barometric pressure) in a summary, as well as "text snippets" for customizing certain vocabulary. For example, the system has seven concepts for different wind levels (CALM, BREEZY, etc.); for each of these there is a standard phrase used to describe it – e.g. [*Winds are*] ***calm*** [*throughout the bay*]. A system administrator or meteorologist could change the word *calm* to *still*, for example, without having to change any of the generator's text planning rules.

4. XML SCHEMAS FOR SENSOR DATA AND REPORT TEMPLATE

ARNS uses XML schema instances (World Wide Web Consortium 2004) to specify the sensor data input and the configurable report templates. These schemas constitute the external interface for the report generator – this not only facilitates integration with other PORTS components, but also defines the range of supported input sensor data in conceptual rather than raw numerical terms.

Figure 2 below illustrates a fragment of the XML schema defining the supported concepts for wind force values: **light**, **normal**, …, **hurricaneForce**, **NA**.

```
<xs:simpleType name="WindForce">
   <xs:restriction base="xs:string">
      <xs:enumeration value="light" />
      <xs:enumeration value="normal" />
      <!-- ... -->
      <xs:enumeration value="hurricaneForce" />
      <xs:enumeration value="NA" />
   </xs:restriction>
</xs:simpleType>
```

Figure 2. Fragment of XML Schema for Wind Force

[4] www.cogentex.com/technology/exemplars/index.shtml

These concept values for sensor data are referenced by the Exemplars text-planning rules, and within the report templates to control their surface rendering. The fixed set of concepts for something like wind force simplifies the design of the generator, and makes the system more modular – in particular, NOAA personnel are able to modify the actual numerical thresholds that define each of these concepts, without modifying anything in the generator or its external interface.

Figure 3 below illustrates an instance of a report template fragment containing:

- a specification of the list of sensor topics to appear in the summary (the **structureSpec** element);
- for each topic, a specification of the attributes to appear in the summary (the **contentSpec** element);
- a specification of the text snippets to be used in the summary (the **textSpec** element). The notation for specifying text snippets is based on the syntax used to specify localizable parameterized message strings in the Java Internationalization API[5] (see *{1}* in Figure 3).

The label 'hurricaneForce' appearing in Figure 3 corresponds to the same label appearing in Figure 2. The XML schemas enforce consistency between these two specifications.

```
<reportTemplate date="2004-11-23-05:00">
  <structureSpec>
    <!-- ... →
    <topic name="windsTopic" />
    <!-- ... →
  </structureSpec>
  <contentSpec>
    <!-- ... →
    <windsTopic force="true" speed="true"
        gust="true" direction="true" />
    <!-- ... →
  </contentSpec>
  <textSpec>
    <!-- ... →
    <windSnippet reported="true"
        snippet="reported winds" />
    <windSnippet reported="false"
        snippet="winds" />
    <!-- ... →
    <windForceSnippet snippet="hurricane force"
        type="hurricaneForce" />
    <!-- ... →
    <temperatureSnippet basis="{1}"
        distribution="within" snippet="in the {1}s" />
    <temperatureSnippet basis="{1}"
        distribution="near"  snippet="near {1}" />
    <!-- ... →
  </textSpec>
</reportTemplate>
```

Figure 3. Fragment of XML Report Template

[5] java.sun.com/docs/books/tutorial/i18n/format/messageintro.html

5. CURRENT STATUS OF ARNS

ARNS components are now fully implemented, but the integration with PORTS is still ongoing. Short text summaries for some individual sensor types can currently be generated from real time observations, but we are still relying on simulated data to test the overall summarization capabilities. Figure 4 shows one of the most complex text summaries generated so far from simulated data, and covering all sensor types.

Observations for San Francisco Bay at 10:30:00 AM PDT on Aug 13, 2004

Water levels are rising at all reporting stations except Redwood City, where they are falling. Levels are above predictions at Golden Gate, Alameda, and Richmond, and below predictions at Port Chicago. All levels are near MLLW. Currents are flooding at all stations in San Francisco Bay, with speeds of 1.6 knots. Reported winds are breezy from the Northeast, except at Alameda, which has light winds. Reported air temperatures throughout the port are near 70 degrees. Water temperatures throughout the port are in the low 60s. Barometric pressure is falling at Port Chicago and Redwood City, and rising at the other stations. Readings are between 1014 and 1016 mb. Reported air gap is 25 feet and increasing at Golden Gate and Alameda, and 23 feet and decreasing at Richmond.

Figure 4. Example of ARNS-generated summary using simulated PORTS data

Such output texts can still be improved in several ways, some of which will be addressed in the final prototype. For example, the word *port* can be replaced by *bay* or *estuary* in the report exemplars for specific locations, where appropriate. Also, some current phrase repetitions will be eliminated to streamline the text. For sentences 6 and 7 in Fig. 4, this would give: *Reported air temperatures throughout the bay are near 70 degrees. Water temperatures are in the low 60s.*

Both simulated data and real-time observation data will be important in testing ARNS before final delivery of the prototype, expected in the first quarter of 2005. Simulated data are especially important for testing the upper linguistic limits of the generator, including detecting any problems with long sentences or the optimal flow of text. Simulations are also needed to test how well the system summarizes and reports on extreme or atypical conditions. Tests with real-time data, on the other hand, will help identify potential operational problems, including what to say (and not say) when certain sensor locations are not reporting, or are providing values or value combinations of doubtful quality.

If the prototype proves robust under extensive testing, NOAA hopes to make ARNS operational at some or all PORTS locations during 2005. In addition, it is anticipated that ARNS would be used to summarize many other CO-OPS products beyond PORTS and to provide data summaries for the exchange of data as partners in the IOOS (Integrated Ocean Observing System).

6. DISCUSSION

ARNS represents an important departure from earlier systems which provide textual summaries of meteorological and oceanographic data. First, it appears to be one of the first uses of natural language generation to nowcasting, as opposed to forecasting. Secondly, since the marine user community has not previously had access to PORTS data in the form of fluent text, there is no "fixed target" of narrative style for the generator to imitate in producing ARNS summaries. Instead, a great deal of control over style is being built into the system, as explained above, so that texts can be rapidly prototyped and then optimized for each user community, and if needed, for each PORTS location.

Our rapid progress in building ARNS is due largely to the modularity, control and relative simplicity made possible by the Exemplars Framework. The elapsed time from project start to generation of summaries as in Figure 4 above has been just three months.

In the context of artificial intelligence approaches to forecasting and nowcasting, it is appropriate to compare ARNS with previous methods for stringing phrases together into text known as "computer worded forecasting" (Glahn 1979), including interactive applications (Ruth and Peroutka 1993). To some extent, the Exemplars approach sketched above is an object-oriented "power tool" for building computer worded reporting systems. Like the earlier approaches, phrase snippets in ARNS are concatenated to build sentences and then whole texts. Linguistic markup is minimal, and oriented to the particular application. The emphasis here, however, is on modularity and maintainability within an XML paradigm, aimed at giving control to users without specialized linguistic training.

ACKNOWLEDGMENTS

The work by CoGenTex on the ARNS generator is supported by NOAA under contract no. EA133C-04-CN-0054. The authors would like to thank Dr. Harry Glahn and his colleagues at the National Weather Service for discussions about their important work in this area over a period of more than two decades.

REFERENCES

Bethem, Thomas and Burton, Janet (2003) The Development of Automated Real-Time Narrative Summaries (ARNS) Using Natural Language Generation Software for PORTS. *Proc. OCEANS 2003 Conference*, San Diego, September 2003.

Glahn, Harry (1979): Computer worded forecasts. *Bul. Amer. Meteor. Soc.*, **60**, 4-11.

Goldberg, Eli, Driedger, Norbert, and Kittredge, Richard (1994): Using Natural Language Processing to Produce Weather Forecasts. *IEEE Expert*, **9**(2), 45-53.

Kittredge, Richard and Lavoie, Benoit (1998) MeteoCogent: A Knowledge-Based Tool for Generating Weather Forecast Text. In *Proceedings of the First Conference on Artificial Intelligence*, American Meteorological Society, Phoenix, 80-83.

Ruth, David, and Peroutka, Matthew (1993): The Interactive Computer Worded Forecast. *Preprints Ninth International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology*, Anaheim, Amer. Meteor. Soc., 321-326.

White, Michael and Caldwell, Ted (1998). EXEMPLARS: A Practical, Extensible Framework for Dynamic Text Generation. In *Proceedings of the Ninth International Workshop on Natural Language Generation*, Niagara-on-the-Lake, Canada, pp. 266-275.

White, Michael and Caldwell, Ted (1999). *Beyond XSL: Generating XML-Annotated Texts with EXEMPLARS*. CoGenTex Technical Report, November 1999.

World Wide Web Consortium. 2004. *XML Schema.* http://www.www.org/XML/Schema