



**WESTERN REGION TECHNICAL ATTACHMENT
NO. 00-05
MARCH 21, 2000**

**SPANISH LANGUAGE WEB PAGE
SECOND GENERATION**

**Miguel K. Miller, Armando L. Garza, Brandt Maxwell, and
Michael Lauderdale - NWSO San Diego, CA**

Introduction

The creation of a Spanish language web page at San Diego, California, was discussed previously (Miller, et al.) in 1999, in a paper which outlined the approach and development of this outreach effort aimed at reaching a large Hispanic community. Shortly after publication and distribution of this document, several offices expressed interest in developing similar web pages for their areas. Since entering into the new millennium, several changes have been made to the initial software programs that have improved the total process and end-state translation received for posting on the Internet.

The purpose for expending valuable resources on this effort remains clear - the demographics of the area have not changed. California is, and will continue to be, a state with a very large resident Hispanic population. As we entered into the new millennium, it is estimated that the California Hispanic population is close to 10 million. This is certainly a more than legitimate driving force for implementation of these types of innovative outreach efforts.

It also comes, as no surprise, that in upgrading the computer being used for the translation process, that tasks for translating products from English to Spanish, have become more efficient. This Technical Attachment (TA) provides a look at the Second Generation translation process that can help offices implement a smarter translation function used for provision of Spanish text products to customers over National Weather Service (NWS) web sites. Using this document eliminates the need to review the old TA, since everything has been incorporated into this paper.

System Requirements

The task for performing the translation from English to Spanish can be accomplished in a variety of ways. Although many innovative methods or approaches may exist, the

following method is used by the San Diego Weather Forecast Office (WFO) to automatically translate issued products from English to Spanish.

Offices desiring to run the translation software should use a dedicated PC. Experience has shown that it is not advisable to run other programs on the same PC since this can result in the disruption of the translation programs. The PC used by the San Diego office has the following specifications: Pentium II 400Mhz PC; 64MB Ram; Windows 98/95; 30MB Hard Disk space; CD-ROM Drive.

A continuously running program is now being used to initiate the translation process. This new development allows for much greater flexibility in translating products. The old method, using a scheduler, limited the types of products to be translated and the times that it could take place. This ruled out the possibility of translating non-routine products or non-routine issuances of routine products. The new program is able to detect when a new product is issued and translate it immediately. Since AWIPS is now the primary source of products, instead of AFOS, the method of sending products to the dedicated Spanish PC has changed. AWIPS sends specified products to LDAD, which upon receipt uses scripts to send the data via FTP over to the PC. On the PC, the products are stored in the "raw" subdirectory, which is for temporary storage. A program called "start" (See Appendix B) continuously searches the "raw" subdirectory for an incoming product which is contained in a predetermined list of products. If it finds a newly inserted product, it moves the product from the "raw" subdirectory into an operational directory and begins the series of translation programs. The "start" program has a valuable "RunWait" command which suspends the search program until the translation process is complete, then resumes the search for more new products.

Translation Software

For many years, offices throughout the country have been experimenting with different types of translation software programs. Offices in Puerto Rico, Florida, and Texas, as well as those in Arizona and California which have high Hispanic populations, have worked hard to identify software that would help the National Weather Service disseminate products to these large communities.

During the past two years, the San Diego office has aggressively worked with a software translation program which is being used to make products available on the office web sites. The setup requires the expertise of someone with a good working knowledge of the Spanish language who can be assigned to work with several software packages (such as WinBatch). If the individual works on this project during slow periods of his/her operational shift and is given a few administrative type shifts, the process to set this up will take 3-4 months. Several more months experiencing different seasonal phenomena will be spent in working with the dictionaries. Any office should then be able to run the software to produce the web site products within 6-8 months.

WinBatch is a Windows-driven software package available at a very reasonable cost and provides the ability to manipulate applications. In San Diego, the WinBatch system, which allows mouse-driven software to be used, is tasked with executing the translation portion of the program using the WBT language.

Two separate software programs are used to accomplish the translations:

- a. **Globalink Power Translator version 6.0.** This is used for text-intensive products such as the Zone Forecast Product (ZFP) and Flash Flood Warnings (FFW). Globalink has been updated to a newer version called L&H Powertranslator Pro.
- b. **Tolken97 Translator version 3.2.** This is used for numeric-intensive products such as Hourly roundups (SWR) and Daily Temperature and Precipitation Summaries (RTP). A new Tolken99 version 4.1 is now available.

Although the processes are similar for both translators which are used for development of the Spanish web page, they will be discussed separately.

Procedures Using Globalink

First, the "start" program searches the "raw" subdirectory which receives AWIPS products for the desired product and moves the product into an operational directory.

Second, a program is run to save the new product in a format which the translator needs for input: the Rich Text Format (RTF). This program (filename.wbt) places the product in a word processor, such as Microsoft WordPad (included with Windows95 and Windows98), and simply saves it as a Rich Text Format file (productname.RTF).

A third program is then run (convfilename.wbt). The WBT language used by WinBatch is designed to execute the following tasks in order: call up Globalink, prepare or "clean up" untranslatable portions of the text, translate the product, save it as a text file (productnamesp.txt), and then exit Globalink. (See Appendix B)

Dots and other symbols found in NWS products inhibit the translator from making a complete translation, so several "find and replace" commands are used to "clean up" the text. Globalink has a deficiency of assuming each line of text as a complete sentence and translates each line as such, often with egregious results. To fix this problem, another "clean up" command was added to remove line breaks. Many solutions to problems actually led to other problems, but most have been solved.

Once the translation is complete, the "start" program starts the fourth program (upfilename.bat) which uploads the file to the web page automatically using a fifth embedded program (filename.scr). Finally, the series of programs are terminated.

This process takes several minutes, depending on the length of the product and the capability of the computer. The longest translation in the suite is the Zone Forecast Product. An average ZFP, for example, will take three to four minutes from start to finish using the aforementioned computer capability.

The "start" program then resumes its search for the next product on its list, and, upon finding a new product, begins the process over again.

Procedures using Tolken97

As with Globalink, the "start" program searches the "raw" subdirectory and moves the product into an operational directory, then begins the other programs.

Tolken97 has no trouble translating the product in its current format, so it is not necessary to convert to Rich Text Format (RTF) by using the "productname.wbt" program as with Globalink.

The "start" program begins the translation using the "convfilename.wbt" file. There is also no need to "clean up" or alter the text in any way. After a few seconds, the translation is complete and the file is saved as "productname.trs", which is then copied to "productnamesp.txt".

The "start" program initiates the upload program (upfilename.bat) with the embedded file (filename.scr) which sends the file to the web page using FTP. The process is finished at this point. (See Appendix C)

This process takes less than one minute. The question that might arise is: "If Tolken97 is so fast, and has no selective formatting preferences, why is it not used to translate all the products?" The answer is that the Tolken97 dictionary is limited and not as robust as the dictionary in Globalink, nor is the grammar syntax as extensive.

Editing Dictionaries

After initial installation, the dictionaries for both Globalink and Tolken97 require much work before they can be used for translating weather terms. They do not include even the most simple meteorological terms. Since the National Weather Service uses a very unique and selective language, most software-provided dictionaries can be described as being "meteorologically challenged".

In order to correct this deficiency, San Diego began with a small glossary of terms provided by the Weather Service Office in San Juan, Puerto Rico. This was very helpful

for many NWS terms, especially marine weather terms such as "Small Craft Advisory", etc. This glossary has been generously augmented to include many more weather vocabulary and NWS-specific terms.

Since other tropical locations do not experience the same types of weather phenomena that California does, it was necessary to make manual translations for such phrases like "Winter Storm Warning." These additional translations do not include every possible weather scenario for every office, but they encompass everything that was listed in the San Diego Station Duty Manual. Much of the editing work has been done. The edited dictionaries and a copy of the previously mentioned unofficial NWS English-Spanish Glossary are quite extensive and can be obtained by contacting Miguel Miller at San Diego.

It should be noted that both translator programs have the ability to translate phrases, or even full sentences, such as a title of a product, e.g., "Extreme Southwestern California Zone Forecast." In Globalink, it is necessary to use all capital letters in the dictionaries while translating phrases so that the final translated text remains in all capital letters.

Tying into the World Wide Web (WWW)

Several modifications were required in order to prepare translated products for dissemination over the web.

First, it was necessary to execute the translation software for a particular product. After receiving a completed translation, identification and correction of meteorologically incorrect translations were required. When new weather scenarios occurred, new words and phrases appeared in the products. Over a period of many months, nearly every weather phenomenon was experienced, and the new words and phrases were added for completion of a reasonably complete dictionary. For example, in southern California, the main weather scenarios are storms during winter, low clouds and fog, Santa Ana winds, and monsoon thunderstorms.

Second, the products were added to the "start" program, translated automatically, checked for problems, but not yet uploaded to the web page. This also provided an opportunity for troubleshooting bugs or glitches which might exist in the WinBatch programs. For example, a time delay command is given after the translation command to give the translator time to complete the translation before saving and exiting. If this time delay is not long enough, WinBatch will continue its commands to save and exit before the translation is complete and disrupt the entire process.

After a trial period of constant vigilance and improvements, the translations became reasonably acceptable and the programs ran smoothly. At this point, these translated products were uploaded and added to the web page. It is important not to add the translated products to the Internet before such a trial period is completed.

As with any computer files, it is recommended that backup files be created for the dictionaries. A failure by the PC will erase all edits. Edits were saved which are necessary for most National Weather Service translations. Each office, however, will need to add their own edits, mainly for local geography nomenclature, localized events, or even specific forecaster writing styles.

Life would be much easier for the translator if all forecasters had good grammar skills and similar styles. Very choppy, incomplete or poor English is also used for the sake of brevity. Since these problems exist, creative solutions need to be made to the dictionaries to accommodate these local translation problems. For example, "Partly cloudy in the afternoon" vs. "Partly cloudy afternoon." Both phrases must be accounted for in the dictionary, using the Spanish equivalent for "in the afternoon."

Concluding Remarks

One of National Weather Services' greatest challenges, in areas where a large number of the residents are Hispanic, is the ability to provide the products and services to these communities in an understandable manner. Whereas the majority of Hispanics are fluent in dual languages, there exists a large number who have difficulty in fully comprehending a more scientific version of the English language. In an effort to make National Weather Service products available to a greater number of customers, an automated system for translating products into Spanish must continue to evolve. These translated products that are now available through office web pages provide a service which reaches out to a previously unreachable population.

This second generation automated translation software is a big improvement over the first version of the translation software that was used for translation of weather products. Efforts to continuously improve translation features are resulting from National Weather Service's commitment to serving extremely large ethnic groups within a County Warning Forecast Area (CWFA). For San Diego, this means reaching out to a large Hispanic customer base where high Hispanic populations reside in Los Angeles County (4.0 million), Orange County (750,000), San Diego County (700,000), and San Bernardino County (530,000). Spanish media and emergency managers in Tijuana, Baja California Mexico also access the web site for planning and decision-making efforts.

Appendix A provides a step-by-step procedure on how to get started using the translation software programs.

Appendix B is an example of how San Diego is using Globalink to translate the zone forecasts.

Appendix C is an example of how Tolken97 translates the more numeric intensive temperature and precipitation summaries.

Finally, it should also be noted that the translation software programs are capable of working with other languages, such as French, German, Italian, and Portuguese. Therefore, depending on the locality of the weather offices and the requirements for serving large, non-English speaking customers, this process can be used in a similar fashion to reach those residents.

To access the San Diego Spanish web page, simply use:

<http://www.wrh.noaa.gov/sandiego/espanol.html>

APPENDIX A

Getting started

1. Dedicate a PC for use in translating the products. (San Diego is currently using a Pentium II 400Mhz PC; 64MB Ram; Windows 98/95; 30MB Hard Disk space; CD-ROM Drive).
2. Install WinBatch, L&H Powertranslator Pro, Tolken99 Version 4.1, and BisonWare FTP Server Version 3.5. Become familiar with their use. Newer versions of translation software have become available since the software was installed at San Diego. The latest versions are mentioned here. The prices were correct as of 11/99.

WinBatch Shareware \$100 1-800-762-8383

email: sales@windowware.com

Internet: <http://www.windowware.com/winware/wb-winbatch.html>

Lernout & Hauspie Powertranslator Pro (formerly Globalink Power Translator)

email: sales@lhsl.com (781) 203-5000 \$150

Internet: <http://www.lhsl.com/powertranslator/>

Tolken99 Version 4.1 (formerly Tolken97 version 3.2) Shareware \$30

email: hagsten@algonet.se Internet: <http://www.algonet.se/~hagsten>

BisonWare FTP Server Version 3.5 Shareware \$25

email: 100416.3553@compuserve.com

Internet: http://ourworld.compuserve.com/homepages/nick_barnes/index.htm

Download and test the shareware items. Once bought, a license is provided to the user.

3. Download backup dictionaries into the translator dictionaries and create your backup.
4. Write programs to download, prepare, translate, and upload the products (see examples in Appendices B and C).
5. Closely monitor numerous translations in a variety of weather scenarios.
6. Edit dictionaries to include local translation needs (Spanish speaker and English-Spanish dictionary are essential).
7. Run the "start" program to automatically run programs.
8. Have a highly motivated individual take responsibility to continually edit dictionaries and ensure the smooth running of the translating programs. **Contact Miguel Miller at NWS San Diego for edited dictionaries.**

APPENDIX B

A Globalink Example Translating the ZFPSGX

The "start" program runs continuously.

A sample of the "start" program appears as follows:

```
:start
if FileExist("c:\products\raw\swrca")
    GoSub swrca
endif
if FileExist("c:\products\raw\cwfs gx")
    GoSub cwf
endif
if FileExist("c:\products\raw\rtps gx")
    GoSub rtp
endif
if FileExist("c:\products\raw\zfps gx")
    GoSub zfp
endif
TimeDelay(10)
Goto start

:swrca
FileMove("c:\products\raw\laxswrca", "c:\download\laxswrca", @false)
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\convswr.wbt")
FileMove("c:\download\laxswrca.trs", "c:\products\laxswrca.trs", @FALSE)
FileMove("c:\products\laxswrca.trs", "c:\download\swrcasp.txt", @FALSE)
RunWait("c:\download\upswrca.bat", "")
return

:cwf
FileMove("c:\products\raw\cwfs gx", "c:\download\cwfs gx", @FALSE)
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\cwf.wbt")
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\convcwf.wbt")
RunWait("c:\download\upcwf.bat", "")
return

:rtp
FileMove("c:\products\raw\rtps gx", "c:\download\rtps gx", @FALSE)
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\convrtp.wbt")
FileMove("c:\download\laxrtp~1.trs", "c:\products\laxrtp~1.trs", @FALSE)
FileMove("c:\products\laxrtp~1.trs", "c:\download\rtps gxsp.txt", @FALSE)
```

```
RunWait("c:\download\uprtp.bat", "")
return
```

```
:zfp
FileMove("c:\products\raw\zfpsgx", "c:\download\zfpsgx", @FALSE)
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\zfp.wbt")
RunWait("c:\Program Files\WinBatch\System\WinBatch.exe", "c:\download\convzfp.wbt")
RunWait("c:\download\upzfp.bat", "")
return
```

GoTo start

The issued ZFPSGX product enters the "raw" subdirectory. The "start" program detects a new ZFPSGX product in the "raw" subdirectory and moves the product file into an operational directory. The zfp.wbt file is run. This file puts the product into a wordprocessor, then saves it as a Rich Text Format (RTF) file so Globalink will translate properly. The language is the WBT language which executes commands by using the keyboard instead of the mouse.

```
Run("c:\progra~1\accessories\wordpad.exe", "c:\download\zfpsgx")
TimeDelay(5)
SendKeysTo("~WordPad", "!f")
TimeDelay(2)
SendKeysTo("~WordPad", "a")
TimeDelay(2)
SendKeysTo("Save As", "zfpsgx.rtf")
SendKeysTo("Save As", "{TAB}")
SendKeysTo("Save As", "r")
SendKeysTo("Save As", "!s")
SendKeysTo("Save As", "y")
SendKeysTo("~WordPad", "!f")
SendKeysTo("~WordPad", "x")
```

Next, the convzfp.wbt file is run. This file (in WBT language) calls up Globalink, opens up the product to be translated, prepares it to be translated by "cleaning up" the formatting glitches, translates it, then saves it as zfpsgxsp.txt.

```
Run("C:\ptwin60\ptwin60.exe", "C:\download\zfpsgx.rtf")
TimeDelay(6)
SendKeysTo("Open.", "~")
TimeDelay(8)
SendKeysTo("Globalink", "!e")
SendKeysTo("Globalink", "r")
```

```
SendKeysTo("Replace","...")
SendKeysTo("Replace","{TAB}")
SendKeysTo("Replace"," ")
SendKeysTo("Replace","!a")
SendKeysTo("Replace","~")
TimeDelay(4)
SendKey("y")
TimeDelay(1)
SendKeysTo("Globalink","~")
SendKeysTo("Replace","T")
SendKeysTo("Replace","{TAB}")
SendKeysTo("Replace"," T")
SendKeysTo("Replace","!a")
SendKeysTo("Replace","~")
TimeDelay(2)
SendKeysTo("Globalink","y")
TimeDelay(2)
SendKeysTo("Globalink","~")
SendKeysTo("Replace",".S")
SendKeysTo("Replace","{TAB}")
SendKeysTo("Replace"," S")
SendKeysTo("Replace","!a")
SendKeysTo("Replace","~")
TimeDelay(2)
SendKeysTo("Globalink","y")
TimeDelay(2)
SendKeysTo("Globalink","~")
```

...similar commands to get rid of ".Monday..." problems.....

```
SendKeysTo("Replace","EXTRE")
SendKeysTo("Replace","{TAB}")
SendKeysTo("Replace"," EXTRE")
SendKeysTo("Replace","!a")
SendKeysTo("Replace","~")
TimeDelay(1)
SendKeysTo("Globalink","y")
TimeDelay(2)
SendKeysTo("Globalink","~")
SendKeysTo("Replace","NATI")
SendKeysTo("Replace","{TAB}")
SendKeysTo("Replace"," NATI")
SendKeysTo("Replace","!a")
```

```
SendKeysTo("Replace", "~")
TimeDelay(1)
SendKeysTo("Globalink", "y")
TimeDelay(2)
SendKeysTo("Globalink", "~")
```

...similar commands to make sure new lines remain intact during translation...

```
SendKeysTo("Replace", "!u")
SendKeysTo("Replace", "{TAB}")
SendKeysTo("Replace", "{TAB}")
SendKeysTo("Replace", "{TAB}")
SendKeysTo("Replace", "{TAB}")
SendKeysTo("Replace", "{TAB}")
SendKeysTo("Replace", "~")
TimeDelay(2)
```

...getting rid of unwanted line breaks...

```
SendKeysTo("Globalink", "^a")
SendKeysTo("Globalink", "!m")
SendKeysTo("Globalink", "p")
TimeDelay(1)
SendKeysTo("Paragraph", !e")
SendKeysTo("Paragraph", "~")
SendKeysTo("Globalink", "y")
TimeDelay(1)
SendKeysTo("Globalink", "{DOWN}")
```

...and now, the translation.....

```
SendKeysTo("Globalink", "^d")
TimeDelay(185)
SendKeysTo("Globalink", "!f")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "{DOWN}")
SendKeysTo("Globalink", "~")
```

TimeDelay(1)

...save the file and exit.....

```
SendKeysTo("Save Translation as","c:\download\zfpsgxsp.txt")
SendKeysTo("Save Translation as",{TAB})
SendKeysTo("Save Translation as",{TAB})
SendKeysTo("Save Translation as",{TAB})
SendKeysTo("Save Translation as",{DOWN})
SendKeysTo("Save Translation as",{TAB})
SendKeysTo("Save Translation as",{TAB})
SendKeysTo("Save Translation as",{~})
SendKeysTo("Save Translation as",{Y})
TimeDelay(3)
SendKeysTo("Globalink",{!F})
SendKeysTo("Globalink",{X})
SendKeysTo("Globalink",{N})
```

Finally, the upzfp.bat file is run which includes the embedded zfp.scr file. These files upload the translated product and insert it into the web page. The ftp address, the username, and the password are uniquely defined by each office.

the upzfp.bat file:

```
@echo off
ftp -n -s:c:\download\zfp.scr
```

the embedded zfp.scr file:

```
open (ftp address)
(username password)
ascii
lcd c:\download
put zfpsgxsp.txt
disconnect
bye
```

APPENDIX C

A Tolken97 Example Translating the RTPSGX

The "start" program detects the new RTPSGX and moves it to an operational directory. Next, the convrtp.wbt file is run. This file (in WBT language) calls up Tolken97, opens up the product to be translated, translates it, saves it as rtpsgx.trs which is then copied to rtpsgxsp.txt by the "start" program.

```
Run("c:\progra~1\tolken97\tolken97.exe","c:\download\rtpsgx")
WinActivate("Tolken97")
TimeDelay(5)
SendKeysTo("Tolken97","^{F7}")
SendKeysTo("Tolken97","~")
TimeDelay(5)
SendKeysTo("Tolken97","^a")
TimeDelay(1)
SendKeysTo("Tolken97","!s")
TimeDelay(2)
SendKeysTo("Tolken97","!f")
SendKeysTo("Tolken97","x")
```

Finally, the uprtp.bat file is run which includes the embedded rtp.scr file. These files upload the translated product and insert it into the web page. The ftp address, the username, and the password are uniquely defined by each office.

the uprtp.bat file:

```
@echo off
ftp -n -s:c:\download\rtp.scr
```

the embedded rtp.scr file:

```
open (ftp address)
(username password)
ascii
lcd c:\download
put rtpsgxsp.txt
disconnect
bye
```