Introduction and Welcome

Welcome to AT&T's FlexTalk SAPI compliant Text to Speech Engine.

FlexTalk provides a powerful and customizable method for converting text to speech while maintaining context sensitive pronunciation, acronyms, proper names, and other aspects of the English language that many text to speech packages stumble over. The FlexTalk engine accepts text as input and applies an extensive list of rules, including those that you add through the use of escape sequences, to convert text to audio output. FlexTalk is not a computer application; it is a software tool that provides text to speech support for a variety of SAPI compliant applications.

FlexTalk offers the following advanced Text to Speech features:

- SAPI Compliance
- Escape Sequences for More Lifelike Speech
- Recognition of Proper Names
- Customizable Voices
- Advanced Handling of Punctuation, Acronyms, and Abbreviations
- Phonetic Transcriptions
- Custom Dictionaries

The FlexTalk Property Sheet allows the user access to settings of the FlexTalk engine. The user can access the Property Sheet either through the Control Panel, or it can be called by an application which is using FlexTalk. The Property Sheet allows the user to create and modify speaker voices and to add, remove or modify word pronunciations. For more information see FlexTalk Property Sheet .

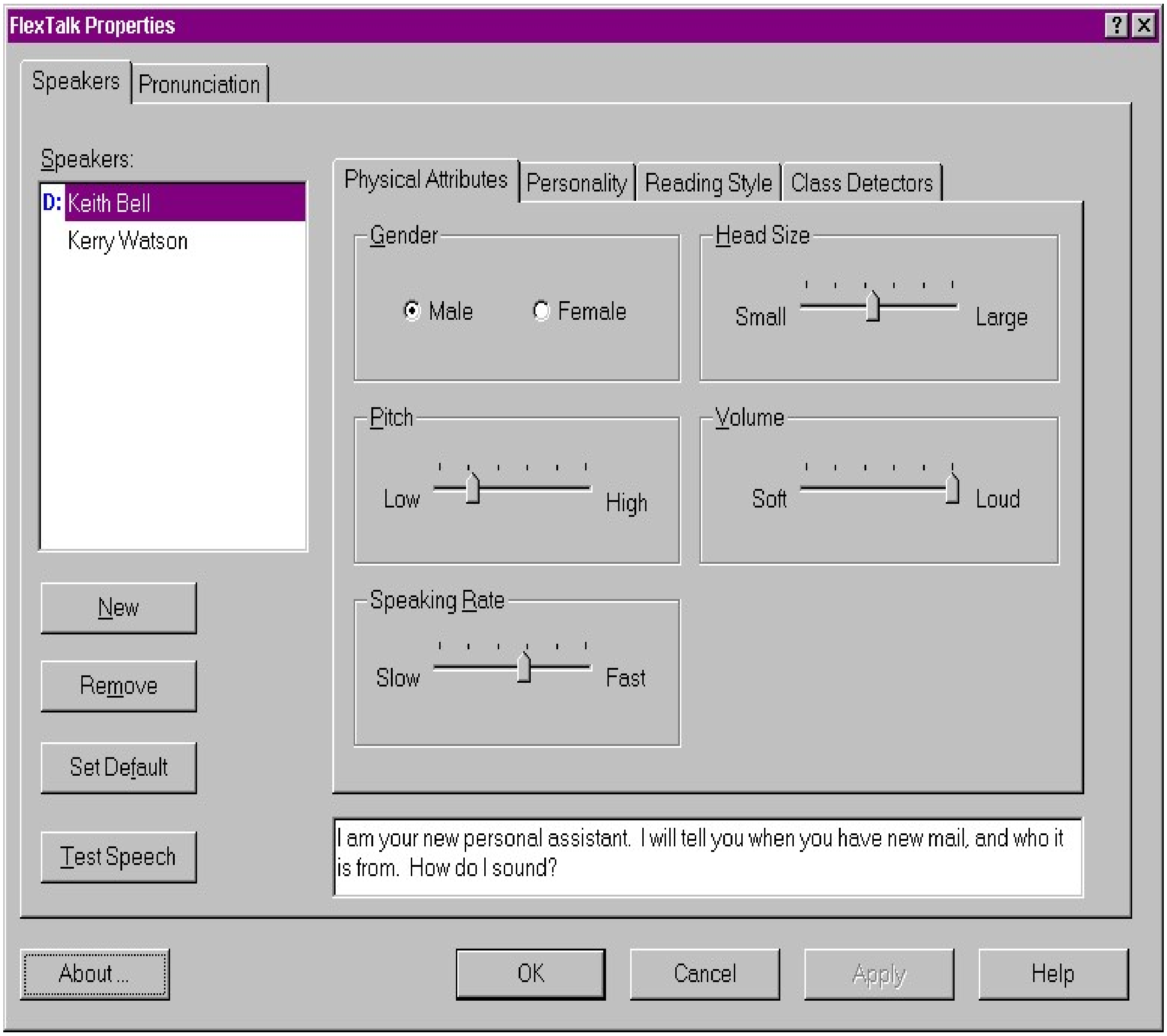
For information on the advanced features of FlexTalk go to Speech Synthesis Control.

For Help on Help, Press F1

FlexTalk Property Sheet

The FlexTalk Property sheet provides the user with access to many of FlexTalk's internal controls. These controls fall into two main categories: those governing the voice and speaking characteristics and those dealing with word pronunciation.

The **Speakers Page** displays information and controls attributes regarding the speakers.



Speakers:
D: Keith Bell
Kerry Watson

Gender: ● Male ○ Female

I am your new personal assistant. I will tell you when you have new mail, and who it is from. How do I sound?

- Speakers list view control - Lists the speaker voices that are currently defined. The settings for the selected speaker are displayed in the tabbed sub-pages. The default speaker is identified by a blue "D:" to the left of the speaker name.
- New button - Allows the creation of a new speaker. When you click on this button, a dialog box is displayed allowing you to specify a name and whether the new speaker is given the default settings, or the settings of an existing speaker.
- Remove button - Allows you to remove a speaker from the list.
- Set Default button - Allows you to set the currently selected speaker as the default speaker.
- Test Speech button - Allows you to test the settings of the currently selected speaker. The text in the edit box next to the Test Speech button is used.

The Physical Attributes sub-page allows the user to set physical attributes such as gender, pitch, and head size. It consists of the following controls:

- Gender - Specifies a male or female voice
- Pitch - Controls the reference pitch setting of the engine

- Speaking Rate - Controls the speaking rate setting of the engine
- Head Size - Controls the front and back vocal tract sizes. The two sizes are controlled in tandem.
- Volume - Controls the volume of the speaker voice.

The Personality sub-page allows you to control the expression and intonation of the speaker. It consists of the following controls:

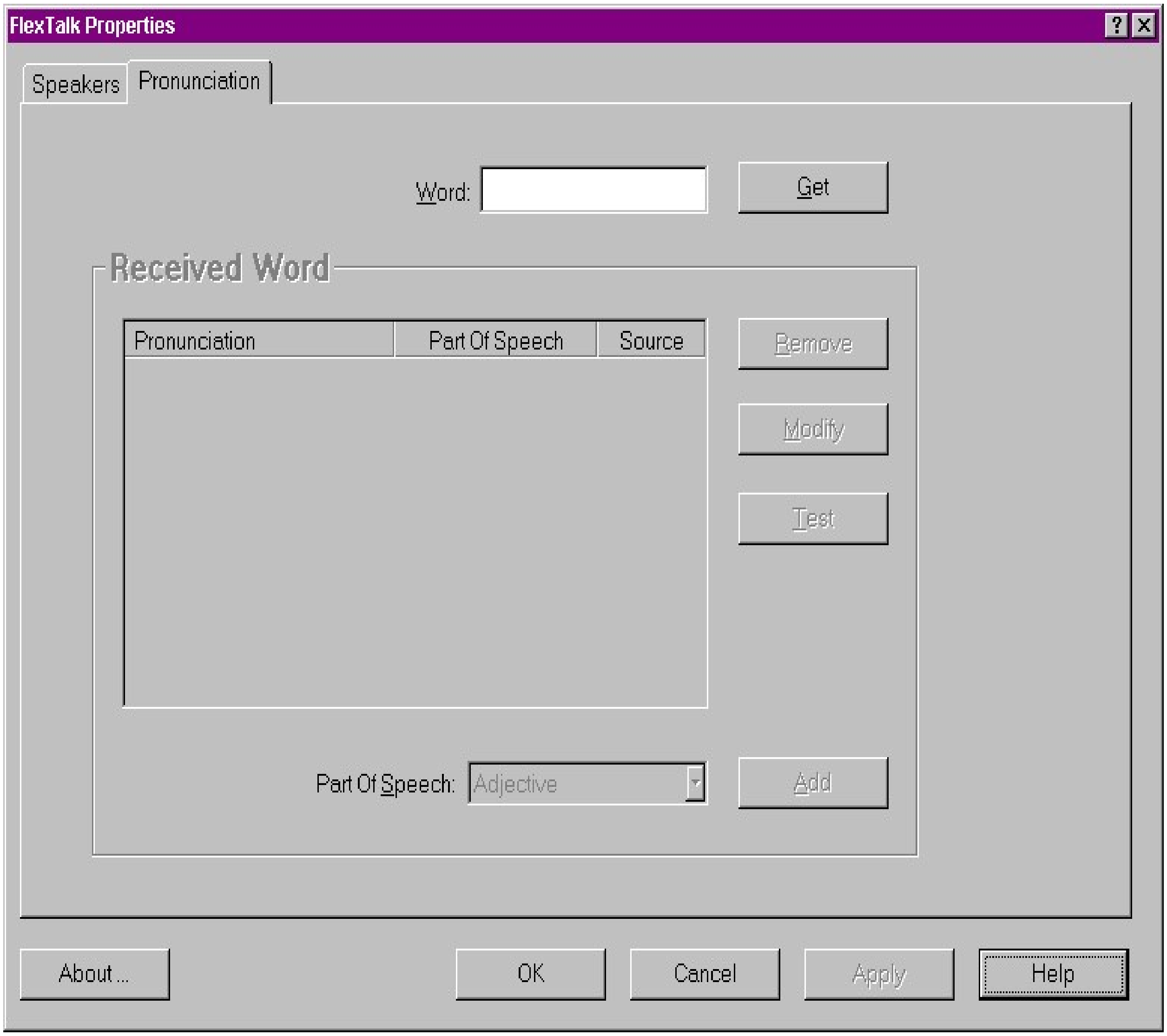
- Expressiveness - Controls the top and base pitches. This setting specifies how much above and below the reference pitch that the top and base pitches are set. The reference pitch is a central pitch for the voice that is defined by the Pitch control on the Physical Attributes tab.
- Mood - This group consists of two controls. The first control specifies the final lowering setting of the engine. The second control specifies the alternating prominences setting of the engine.

The Reading Style sub-page controls the way FlexTalk reads text. It consists of the following controls:

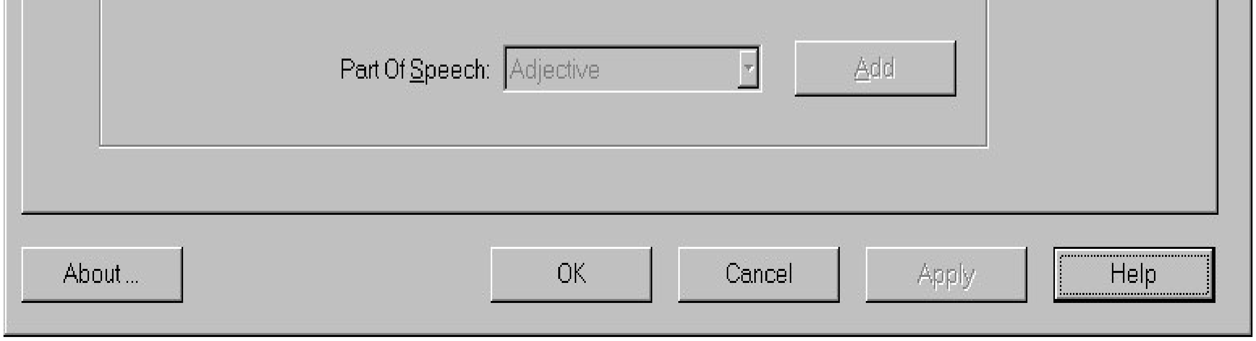
- Hyphens - Controls the alphabetic and numerical hyphen modes.
- End of Sentence Terminator - Controls whether the end of a sentence is indicated by the end of a text line or by normal punctuation.
- Words in upper case letters - Controls the case significance/insignificance mode.
- Abbreviations - Controls the abbreviations mode
- Mode of speech - Allows you to set the spell, proofread, and math modes.

The Class Detectors sub-page controls the FlexTalk class detectors. It consists of combo box with the settings Off, On, and Risky for each of the class detectors.

The **Pronunciation Page** allows the user to modify the TTS dictionary. It allows the following actions to be performed by the user:



Part Of Speech: Adjective



Getting the Pronunciations of a Word

Type the desired word in the Word edit box and click on the Get button. This fills the list view control with the default pronunciation of the word and the user-defined pronunciations, if any (no more than one for each part of speech).

Modifying the Pronunciation of a Word

You can modify the pronunciation of a word either by selecting the pronunciation and clicking on the Modify button, or by double-clicking on the pronunciation. You can cancel the editing operation by pressing escape. Note that only user-defined pronunciations may be modified.

Adding a Pronunciation of a Word

You can add a pronunciation of a word by selecting the desired part of speech in the Part Of Speech combo box and then clicking on the Add button. Only the parts of speech that do not already have pronunciations are displayed. Therefore, if a pronunciation for the Noun part of speech already exists, another Noun pronunciation cannot be added. Instead you must modify the pronunciation for the Noun.

Removing a Pronunciation of a Word

You can remove a pronunciation by selecting the pronunciation and clicking the remove button. Only pronunciations in the user exception dictionary can be deleted.

Testing a Pronunciation of a Word

You can test the pronunciation of a word by selecting the pronunciation and clicking on the Test button. For more information on Cecilbet phonemes, see The CECILbet Phonetic Alphabet .

Go Back To Introduction and Welcome

Speech Synthesis Control

Escape Sequences

FlexTalk accepts a number of special input strings, or Escape Sequences , which change the resulting output speech. They are commonly used to fine-tune particular sentences for specific applications such as perception tests. Vowels can be shortened, pauses lengthened, stresses changed, etc.

Basic Translation Rules

There are two types of text translation rules: the Basic Translation Rules and the Text Class Rules . The Basic Translation Rules handle things such as end of sentence detection, punctuation expansion and abbreviation expansion. These rules are usually simple and do not use context beyond the immediately neighboring words to disambiguate the translation. Also, they are internal, that is, they are hard coded within FlexTalk and cannot be easily changed. There is a limited way to control some Basic Rules. For examples the expansion of abbreviations can be turned off or on, or an end of sentence can be forced using an escape sequence. Also frequently the abbreviation expansion rules use the settings of the Text Class Detectors (off, conservative or risky) to determine how (and if) to expand abbreviations.

Text Class Rules

The Text Class Rules attempt to detect certain classes of text (addresses, phone numbers, etc.) and perform translations appropriate for the class. For example when a telephone number is detected, each digit should be read individually (in serial mode) with pauses inserted at appropriate places. The Text Class Rules are generally more complex than the basic rules and look at a number of words when trying to detect syntactic patterns.

Special Translation Modes

FlexTalk provides four Special Translation Modes . They are turned on and off using escape sequences as shown below. If more than one mode is on at the same time, the first one on the list below is used. If none of the special translation modes is on, FlexTalk is in a standard translation mode.

Phonemic Input

It is also possible to explicitly specify pronunciations by inserting phoneme strings in the input text. Phonemes should be those defined in the CECILbet Phonetic Alphabet. Phoneme strings are placed between \(and \) delimiters. Double quote places primary stress on the following vowel. Hyphen forces syllabification.

Examples: This is an \(ig"zam-p&l \) of embedded phonetics.
My name is \("m @ r k "byUt.nA-g&l \) .

Original orthography can be supplied within curly brackets {}, and is highly recommended because some TTS modules depend on it.

The \({test} t"est \) was \({silly} "sil-E \) .

Explicit durations can be supplied for any phoneme...
\(.O[75]_kla"hO-ma \)
where the number in [] gives the duration of the previous phone in 100ths of a second, i.e. "O[100]" is one second.

Go Back To Introduction and Welcome

Escape Sequences

FlexTalk accepts a number of special input strings, or escape sequences, which change the resulting output speech. They are commonly used to fine-tune particular sentences for specific applications such as perception tests. Vowels can be shortened, pauses lengthened, stresses changed, etc.

Note that each entire sequence must be surrounded by some white space (or the beginning of a line) and not mixed in with, among other things, punctuation.

Escape Sequences for Text Analysis Control:

\!n<class><mode>

where class is one of:

- a - addresses
- d - dates
- f - fractions
- i - measurements
- m - money
- n - proper names
- o - ordinal numbers
- p - telephone numbers
- t - times

and mode of the class detector is one of:

- o - off
- c - conservative
- r - risky

\!sb ... \!se_	Spell Mode Begin/End
\!mb ... \!me	Math Mode Begin/End
\!rb ... \!re	Raw Mode Begin/End
\!pb ... \!pe	Proofread Mode Begin/End
\!ab ... \!ae	Acronym Spell Mode Begin/End
\!eb ... \!ee	Abbreviation Expansion Mode Begin/End
\!hnv ... \!hnt	Set Numerical Hyphen Verbose/Terse Mode
\!hav ...\!hat	Set Alphabetic Hyphen Verbose/Terse Mode
\!br	Force End of Sentence
\!cb	Switch to Case Significant Mode
\!ce	Switch to Case Insignificant Mode
\!co	Show Current Setting of Text Class Detection and Text Expansion Modes
\!de	Reset All Text Class Detection and Text Expansion Modes to Defaults
\!pn	Identifies the following word as a proper noun

Escapes NOT Presuming Phrasing:

\!r<rate>, \!R<rate> sets speaking rate. <rate> is in the range 0.1 - 2.0, with 1.0 default, 1.2 for 20% slower, 0.5 for 50% faster, etc. The strings "normal", "fast", "slow", "fastest" and "slowest" are also recognized. (Roughly, "slowest" corresponds to about 85 words per minute; "normal" to about 190 words per minute; "fastest" to 355 words per minute.) The 'R' form changes rate permanently; the 'r' form only for the current major phrase

\!-	de-accent the following word
\!*<tone><prominence>	accent following word with tone L*, H*, L+H*, L*+H, H+L*, or H*+L and optional prominence value
\!!*<tone><prominence>	as above, but also de-accents subsequent words in the orthographic minor
\!c	cliticize the following word: i.e., de-accent it and remove any duration-influencing stresses
\!#<emphasis>	set emphasis of the following word
\!|<emphasis>	set emphasis of the boundary following the preceding word
\!_	down step every non-initial pitch accent in the orthographic minor
\!@	down step the next word

\!f insert a floating phrase accent

\!%<pitch>, \!%%<pitch> set pitch scaling; 1.0 is the normal scaling; '%%' form sets pitch permanently, '%' only for current major phrase

\!C<string> place <string> in the comment field of the next real word

Escapes Presuming Phrasing:

\!{<param><value> Set Phrase Parameters with Reset

The settings apply only to the current major phrase and are reset to prior settings after the phrase is spoken.

\!{!<param><value> Set Phrase Parameters without Reset

The new settings are retained after the current phrase is spoken.

sets "param" for this minor phrase, where "param" is one of the following F0 phrasal parameters with defaults (<male>, <female>):

T - minor top line (115Hz, 195Hz)
R - minor reference line (96 Hz, 165Hz)
B - minor base line (75Hz, 140Hz)
X - alternating prominences ... (60%, 60%)
P - alternating prominences ... (50%, 50%)
K - major downstep ratio (60%, 60%)
F - major final lowering ratio (90%, 90%)
C - major rate scalar (75%, 75%)
D - ratio for H* legs (40%, 40%)
E - major final raising ratio (120%, 120%)
G - end effect time (50%, 50%)
U - time fraction for H- (30%, 30%)

\!p<tone><prominence> Set phrase accent for current minor phrase; "tone" is high (H) or low (L) *****need value definitions.

\!i<tone><prominence> Set initial boundary tone for current minor phrase; "tone" is high (H) or low (L)

\!b<tone><prominence> Set final boundary tone for current minor phrase; "tone" is high (H) or low (L)

\!s<position><csec> Add silence to minor phrase at position initial (i) or final (f); interval in centiseconds

\!? Put a yes/no question intonation pattern on the current minor

\!/ Cause a continuation rise at the end of the current minor

\!us<gender>, \!uS<gender> Set speaker's gender to male (m) or female (f); use 's' to change current minor phrase, 'S' for permanent change

\!u<position><scale> Scale back (b) or front (f) of vocal tract for this minor; use B or F to change scale permanently. 1.0 is default; > 1.0 is larger; < 1.0 is smaller.

\!wH<value> Controls voice aspiration (“breathiness” or “coarseness”). Values range from 0.0 upward, but larger than about 5.0 becomes unintelligible.

\!WS<value> Controls spectral tilt; larger values make the voice sound hushed, almost whispering.

Other Escape Sequence

Listed below are the remaining escape sequences and their functions.

\!br force end of sentence regardless of punctuation,
\!cb switch to case significant mode (default),
\!ce switch to case insignificant mode,
\!de set all text class detection modes and text expansion modes to default,

Go Back To Speech Synthesis Control
Go Back To Introduction and Welcome

Basic Translation Rules

Changing Text Expansion Modes

The escape sequences are:

\!ab	begin acronym spell mode (default),
\!ae	end acronym spell mode
\!eb	begin abbreviation expansion mode (default),
\!ee	end abbreviation expansion mode,
\!hnv	set numerical hyphen verbose mode (default)
\!hnt	set numerical hyphen terse mode
\!hav	set alphabetic hyphen verbose mode,
\!hat	set alphabetic hyphen terse mode (default),
\!xb	begin detection of externally defined text classes (default),
\!xe	end detection of externally defined text classes.

Abbreviation Expansion

The first set of basic rules and expansions applied to the incoming text is abbreviation expansion. Each word (except numbers and punctuation) is compared against a list of all known abbreviations. If a match is found, the table provides information about the abbreviation class of this abbreviation. Given the class, FlexTalk performs a particular set of tests to determine if and how to expand the abbreviation.

If an abbreviation has at most one possible expansion and does not require any special handling (for example hr to hour), it belongs to one of the common abbreviation classes. If the abbreviation requires application of special rules to determine if and how to expand it, it belongs to its own special abbreviation class with its own special rules. There are several reasons why an abbreviation may require its own special abbreviation class: it has more than one expansion (for example fl to Florida, floor, fluid or falls), or it is also a common word (for example apt can be apartment or apt), or it requires some type of special handling that is not provided by any of the common classes.

All abbreviations can be followed by an optional period which is either deleted or kept as an end of sentence period. If an abbreviation is within a matched pattern that includes word(s) that follow the abbreviation, the period is obviously not an end of sentence and is deleted. For example, personal titles requires that the abbreviation be followed by at least one word (proper name). In other cases, an end of sentence algorithm is applied to determine what to do about the period. See the **End of Sentence Detection** below. Note, also, that expansion of some abbreviation classes depends on class detector settings, such as address risky mode.

Expansion of an abbreviation can be inhibited by enclosing the text in double quotes. In addition, expansion of abbreviations can be enabled or disabled through the FlexTalk Property Sheet (Reading Style page). Expansion of all abbreviations can also be turned off via the escape sequence \!ee and turned on (default) via \!eb.

End of Sentence Detection

End of sentence detection can be driven either by normal sentence punctuation or by the end of a text line (CR character). End of sentence punctuation includes: .!? (period, exclamation point and question mark). Selection of these modes can be accomplished through the FlexTalk Property Sheet (Reading Style page). Modes can also be selected through the use of control tags. End of line mode is selected by the tag \Ctx=“line”\. Punctuation mode is selected by the tag \Ctx=“sentence”\.

Acronyms and Alphanumerics

An acronym contains only all upper case letters that is not an abbreviation. An alphanumeric contains both letters and numbers. If the acronym spell mode is on, all acronyms and the alphabetic portion of the alphanumeric are spelled out (separated by spaces). If the acronym spell is off, these alphabetic fields are not changed and the text synthesizer may pronounce them as a word. Refer to the Numbers below for expansion of numbers within alphanumerics. The acronym spell mode can be controlled through the Reading Style page of the FlexTalk Property Sheet . It can also be turned on (default) via the \!ab escape sequence and turned off via the \!ae escape sequence.

In the case insignificant mode, acronyms are not recognized. To be pronounced correctly, they must be either in the abbreviation dictionary, or the letters must be separated by spaces or periods. Some examples are i.b.m. or i b m.

Numbers

Number expansion is performed after the text class detection and expansion and before handling punctuation. There are four ways (modes) in which FlexTalk can expand a number.

Serial: each digit is translated as a single word. For example 1234 expands into one two three four.

Comma: the number is read with all place quantifiers made explicit. For example 1234 expands into one thousand two hundred thirty four.

Pair: each pair of digits is grouped and translated as a unit. For odd length strings, the first digit is translated alone. For example 1234 expands into twelve thirty four and 12345 expands to one twenty three forty five.

Hundreds: the first two digits are grouped together and translated together, followed by their place quantifier. This mode only applies to numbers in range 1100-9999 (e.g. dates). For example 1234 expands into twelve hundred thirty four.

FlexTalk contains a complex set of rules that select between these modes based on the context in which the number appears. These rules enable FlexTalk to reliably speak numbers in many settings, such as phone numbers, dates, and addresses. While there is no direct user control over these interpretations, some elements may be controlled through the Class Detectors page of the FlexTalk Property Sheet .

Roman Numerals

The recognized roman numeral must contain more than one letter, must conform to the roman numeral syntax and must only contain letters I, V, X, i, v or x. The highest recognized numeral is XXXVIII, the lowest is II. The lower case numerals are only recognized when in the case insignificant mode. There is no way to inhibit expansion of roman numerals.

Examples:

Valid numerals: II, XIV, XXXVIII

Invalid numerals: X (single letter), IXV (syntax), XXXX (syntax), LX (unrecognized letter).

Unrecognized numerals are either spelled out as acronyms (if all capital letters) or left unchanged.

Once detected, roman numerals are expanded in two ways.

1) If the numeral is detected as a part of a name (names text class) or the name risky mode is set, the numeral is expanded as an ordinal number, for example: Henry VIII as Henry the eighth.

2) In all other cases the numeral is expanded as cardinal number, for example: Chapter IV as Chapter four.

Punctuation

Punctuation characters are: {}!@#$%^&*()-_=+[]\|`~;:'",<>./?. Punctuation expansion is one of the last steps that FlexTalk performs. The rules described in this section apply only to the punctuation that was not handled in one of the previous steps (for example by a text class detector or during number translation).

There are three ways in which FlexTalk handles punctuation:

1)Punctuation delimiting the end of sentence is passed to the text synthesizer unchanged.

2)The punctuation characters: -"'().%#& (hyphens, quotes, parentheses, periods, percents, pound signs, and ampersands) that occur within specific syntax are translated using special rules which are described below. This special handling takes effect only if none of the special translation modes is set (math mode, spell mode, etc.).

3) Punctuation that does not fit into one of the above two categories is translated to its word equivalent (for example & to

ampersand) or left unchanged, depending on the context. The translations depend on the current translation mode (generic mode, math mode, spell mode, etc.).

Hyphens

The hyphens are handled by classifying them into three categories as listed below:

1) Word hyphenation hyphen. Such hyphens must be immediately preceded by a word (no space), and must be immediately followed by a newline and a word. The hyphen is deleted from the word.

2) Numeric Hyphen. A hyphen is classified as numeric if it is not immediately followed by another hyphen (with no intervening space) and matches one of the two rules below:

- preceded by word and space(s) and immediately followed by a number, punctuation, alphanumeric or a plain word, for example value of -1, or

- preceded by a number or punctuation or alphanumeric, optional spaces(s) and hyphen(s) and followed by optional space(s), hyphen(s) and number or punctuation or alphanumeric, for example 3B2-20.

3) Alphabetic hyphen. A hyphen is classified as alphabetic if it does not fall in any of the above categories.

Although the expansion rules for the numeric and alphabetic hyphens are essentially identical, the default behavior is to expand numerical hyphens as dash and ignore the alphabetic hyphens.

Quotes

Use of quotes (single or double) will have cause few unexpected results. Contractions will be handled correctly and enclosing words in quotes will cause either a trailing pause (single quotes) or emphasis (double quotes). Mismatched quotes and leading quotes may sometimes cause FlexTalk to pronounce the word “apostrophe” or “quote” when confusing punctuation is encountered.

There is no direct user control over FlexTalk’s handling of quotes. However, some class detectors, especially measurement risky mode will affect interpretation.

Parentheses

A left parenthesis is replaced with a comma (pause) if it is the first (outside) one in a parenthesized expression and if it is preceded by a space or quotes. In other cases it is expanded to the words “open paren” or “left parenthesis” (Math Mode).

A right parenthesis is replaced with a comma (pause) only if it has a previous matching left parenthesis which was replaced by a comma. In other cases it is expanded to the words “close paren” or “right parenthesis” (Math Mode).

For example:

printf() function	printf open paren close paren function
see printf (3c) in the	see printf , three c , in the
chapter 3 (section 6(c))	chapter three , section open paren c close paren ,
nested ((twice)) are	nested , open paren twice close paren , are

Other Punctuation

A period that is immediately followed by a number is translated to point. A percent sign that is preceded by a number and an optional space is translated to percent. A pound sign that is immediately followed by a number or alphanumeric is translated to number. Ampersand is always expanded to and. In all other cases, the expansion is to the word “ampersand”.

Go Back To Speech Synthesis Control
Go Back To Introduction and Welcome

Text Class Rules

There are nine instances of text classes that are detected by FlexTalk: addresses, dates, fractions, money, telephone numbers, time, ordinal numbers, measurements and names. Each one of these "detectors" can be set in one of the three modes (Text Class Detection Modes): off, conservative and risky. Initially, all the detectors are in the conservative mode. Refer to the Text Classes for description of how these modes operate. The text class detection modes can be changed using escape sequences with the following general format:

\!ncm

where c is one of the following text classes:

a	for address class
d	for date class,
f	for fraction class,
i	for measurements class,
m	for money class,
n	for proper names class,
o	for ordinal numbers class,
p	for telephone numbers class,
t	for time class,

and m is one of the three possible class detector settings for this text class:

o	turn the class detector off,
c	set the class detector to conservative mode (default),
r	set the class detector to risky mode.

Examples:
Switch address detector to risky mode:
\!nar

Switch date detector to conservative mode:
\!ndc

Switch telephone numbers detector off:
\!npo

ADDRESSES Text Class

In the address conservative mode (default, also set via the \!nac escape), only full addresses are detected. These can have one of the following formats. Optional punctuation is omitted for clarity.

Examples:

[Name][Location]-Street-[Location]-CityState-	Roger Rabbit 1 Main Street Podunk NJ
[Name]-PoBox-number-CityState-	Elsie Borden PO Box 101 Alta UT 89001

In the address risky mode (set via the \!nar escape), both full addresses described above and partial addresses below are detected. These can have one of the following formats.

[Name][Location]-Street-[Location]	Dr Smith, 1 Smith Dr
-PoBox-number-	P.O. Box 120001
-CityStateZip-	Yorkville Illinois 60560

TELEPHONE NUMBERS Text Class

In the phone conservative mode (default, also set via the \!npc escape), the following formats of telephone numbers are detected.

Examples

AreaCode-(3number)-hyphen-(4number)-[Extension]	1-708-713-5290
(3number)-hyphen-(4number)-[Extension]	555-1212 x3550

In the phone risky mode (set via the \!npr escape), both the conservative syntax above plus the following syntax are detected.

AreaCode-7(word|acronym)-	1 (800) CALLATT
AreaCode-3(word|acronym)-hyphen-4(word|acronym)-	1-900-get-help
(3number)-(4(word|acronym))-	345-NEWS
nOneOne-	911

FRACTIONS Text Class

The fractions are detected and expanded only in the fraction risky mode (set via the \!nfr escape). The following fraction syntax is recognized.

Examples

FractPart-	3/4
number-[hyphen]-FractPart	1-1/2

TIME Text Class

In the time conservative mode (default, also set via the \!ntc escape), the following syntax of time is detected.

Example

Time-["o'clock"]-AmPm-	3:10 a.m.

In the time risky mode (set via the \!ntr escape), both the conservative syntax above plus the following syntax are detected.

Time-["o'clock"]	9:27 o'clock

DATES Text Class

In the date conservative mode (default, also set via the \!ndc escape), the following syntax of time is detected.

Examples

MonthName-(number<32)-[OrdEnd][MddRange][Year]	Dec 24th-26th 1990
[DdmRange]-(number<32)-[OrdEnd]-MonthName-[Year]	24 Dec.-26 Dec. 1990

In the date risky mode (set via the \!ndr escape), both the conservative syntax above plus the following syntax are detected.

(number<13)-"/"-(number<32)-[Year]	12/24/90
YearRisky	'09

MONEY Text Class

In the money conservative mode (default, also set via the \!nmc escape), the following syntax of monetary amounts is detected.

Examples

"$"-[period]-number-[period][number]-Cents-	$.25 cents
"$"-LargeAmt-[LargeRange]-Zillions-[Dollars]	$1.2 million - 2 million
"$"-SmallAmt-[SmallRange][Dollars]	$1 dollar

In the money risky mode (set via the \!nmr escape), the same syntax as above is detected, except the initial dollar sign is not required. This in effect matches any number as a dollar amount.

PROPER NAMES Text Class

In the name conservative mode (default, also set via the \!nnc escape) the names are detected only when case is not significant (set via the \!ce escape). The reason is that when case is significant, the recognition and correct pronunciation of proper names can be achieved by capitalizing them. However, when case is being ignored, FlexTalk must use a riskier strategy to determine proper names. In this mode the following syntax of names is recognized. Optional punctuation is omitted for clarity.

Example

TitleCons-[FirstRisky][word][PersEpithet][roman]	mr. jones jr. iii
FirstCons-word-[PersEpithet][roman]	j. b. jones

In the name risky mode (set via the \!nnr escape), the following syntax of names is recognized. Optional punctuation is omitted for clarity.

Examples

(word|letter)-[word|letter][And][word|letter]-CoEpithet	Brown & Green Inc.
TitleRisky-[FirstRisky]-word-[PersEpithet][roman]	Mr J Jones Sr.
TitleRisky-FirstRisky-[PersEpithet][roman]	St John
FirstRisky-word-[PersEpithet][roman]	John Jones
roman-	XXII

MEASUREMENTS Text Class

The measurements are only detected in the measurement risky mode (set via the \!nir escape). Any number immediately followed by an unmatched single or double quote is a measurement and is expanded as follows.

- If the number preceding the single or double quote is an expanded fraction with no whole number, the quote is expanded to of a foot or of an inch respectively. For example, if measurement and fraction risky modes are set, 1/2" expands to one half of an inch, and 3/4' expands to three fourths of a foot.

- If the number preceding the single or double quote is an expanded fraction containing a whole number, the quote is expanded to feet and inches respectively. For example, if measurement and fraction risky modes are set, 1 1/2" expands to one and one half inches, and 3 3/4' expands to three and three fourths feet.

- If the number preceding the single or double quote is 1, the quote is expanded to foot or inch respectively. For example 1" expands as one inch.

- In all other cases, the single or double quote is expanded to feet or inches respectively. For example 17' expands to seventeen feet, or 1/2" expands to one slash two inches.

ORDINALS Text Class

In the ordinal conservative mode (default, also set via the \!noc escape), the ordinal ending must match the preceding number and it must follow the number without an intervening space. The possible ordinal endings are st, nd, rd and th. For example 3rd is detected as ordinal number, but 3th or 3 rd is not. If an ordinal number is detected, the ordinal ending is deleted and the number is translated as an ordinal, for example 27th is translated as twenty seventh. If an ordinal number is not detected, the ordinal ending is left unchanged, for example 3th is translated as three th.

In the ordinal risky mode (set via the \!nor escape), the ordinal ending does not have to match the preceding number, but it still must follow the number without an intervening space. The same translation rules are used as in the conservative mode. For example 3th is translated as third, and 5 th is translated as five th.

Go Back To Speech Synthesis Control
Go Back To Introduction and Welcome

Special Translation Modes

For several of the special modes, FlexTalk uses a citation intonation. This intonation pattern treats each word as a separate phrase and pronounces the word as if it was a one word sentence. The default is generic mode, which in itself is not a special translation mode.

The escape sequences are:

\!sb	spell mode begin
\!se	spell mode end
\!mb	math mode begin
\!me	math mode end
\!rb	raw mode begin
\!re	raw mode end
\!pb	proofread mode begin
\!pe	proofread mode end

Raw Mode

Raw mode produces a literal dump of the input text, character by character. For example, Hello world. is translated as "cap, h, e, l, l, o(.) space(.) w, o, r, l, d(.) period(.)". In the previous example, the sequence (.) marks the intonation change denoting an end of sentence.

The raw mode uses the following translation rules:

- every ASCII character (including whitespace and period) is translated,
- all unprintable characters are translated,
- numbers are translated serially,
- embedded escape sequences are translated as well as activated,
- upper case characters are preceded by cap,
- intonation is set to citation.

Spell Mode

Spell mode spells out a section of text. For example Hello world. is translated as "h, e, l, l, o(.) w, o, r, l, d(.) period(.)". In the previous example, the sequence (.) marks the intonation change denoting an end of sentence.

The spell mode uses the following translation rules:

- all words are spelled,
- all numbers are translated serially,
- all punctuation is translated (pronounced as words, such as “period” or “comma”),
- unprintable characters are ignored,
- embedded escape sequences are activated but not translated,
- lower and upper case letters are not distinguished,
- intonation is set to citation.

Proofread Mode

Proofread mode is similar to standard translation mode, except it uses the citation intonation and is more explicit in some cases. For example Hello world. is translated as "hello(.) world(.) period(.)". In the previous example, the sequence (.) marks the intonation change denoting an end of sentence.

The proofread mode uses the following translation rules:

- most words are passed through for pronunciation,
- words with embedded punctuation are spelled out,
- numbers are translated in serial mode,

- all punctuation is translated (pronounced as words, such as “period” or “comma”).
- unprintable characters are ignored,
- embedded escape sequences are activated but not translated,
- single upper case letters and upper case letters in acronyms are preceded by cap,
- intonation is set to citation.

Math Mode

In Math Mode all punctuation is expanded and some punctuation is interpreted as mathematical symbols. For example 1+2-3/4=x is translated as "one plus two minus three divided by four equals x". A newline always ends a sentence. A period does not end a sentence, but is expanded as point.

Go Back To Speech Synthesis Control
Go Back To Introduction and Welcome

The CECILbet Phonetic Alphabet

Main Stress: (") precedes syllable
Secondary Stress:(.) precedes syllable
Reduced Stress: (_) precedes syllable

Embedded hypens clarify syllable boundaries without specifying stress.

Example:
justification .J^st&f&"kAS&n

Symbol	Example	Transcription	Symbol	Example	Transcription
E	heed	"hEd	m	ham	"ham
i	hid	"hid	n	nip	"nip
e	head	"hed		span	"span
a	had	"had	N	hang	"haN
@	father	"f@DR	b	bit	"bit
o	hot	"hot	d	dip	"dip
u	hook	"huk	g	got	"got
U	rue	"rU	p	pik	"pik
>	saw	"s>		spit	"spit
O	hoe	"hO	t	tip	"tip
&	data	"dAt&		stick	"stik
^	cup	"k^p	k	kick	"kik
~	little	"lit~l		skip	"skip
R	her	"hR	v	voice	"vYs
I	hide	"hId	D	this	"Dis
Y	boy	"bY	z	zap	"zap
A	hay	"hA	Z	measure	"meZR
W	how	"hW	f	fool	"fUl
r	rat	"rat	T	thick	"Tik
	tar	"t@r	s	sick	"sik
l	lid	"lid	S	ship	"Sip
	hall	"h>l	h	hit	"hit
w	we	"wE	J	judge	"J^J
y	yes	"yes	C	chip	"Cip
m	moo	"mU	*	silence	“*”

Go Back To Speech Synthesis Control
Go Back To Introduction and Welcome