
Language Supplement



Speechify 3.0 for en-US
(US English)

Document History

Date	Release Name
August 2003	Third Edition – for Speechify 3.0
February 2003	Second Edition, Update 1 – for Speechify 2.1.6
January 2003	Second Edition – for Speechify 2.1.5
September 2002	First Edition, update 4 – for Speechify 2.1.3
June 2002	First Edition, update 3 – for Speechify 2.1
March 2002	First Edition, update 2 – for Speechify 2.x
March 2002	First Edition, update – for Speechify 2.0
January 2002	First Edition

Notice

Copyright © 1995–2003 by SpeechWorks International, Inc. All rights reserved.

The information in this document is subject to change without notice.

U.S. Patent Nos. 5,995,928; 5,809,494; 5,765,130; 6,061,651; and 6,173,266. One or more patents may be pending in the United States and other countries.

SpeechWorks, DialogModules, SADL, SMARTRecognizer, SpeechCare, SpeechCookie, Speechify, SpeechSecure, SpeechSite, SpeechSpot, SpeechWorks Here, the SpeechWorks logo and the SpeechWorks Here logo are trademarks or registered trademarks of SpeechWorks International, Inc. in the United States and other countries. All other trademarks are property of their respective owners. Windows NT is a registered trademark of Microsoft Corporation.

Portions of Speechify software are subject to copyrights of GlobeTrotter Software, Inc.

Published by:

SpeechWorks International, Inc.
695 Atlantic Avenue
Boston, MA 02111

Table of Contents

1. Introduction

Overview	1
New and changed information	1
Getting started	2

2. Standard Text Normalization

Numbers	4
Cardinal numbers	4
Floating point digits	5
Negative numbers	5
Fractions	5
Zero-initial numbers	6
Numeric expressions	7
Dates	7
Days of the month	8
Times	9
Currency	9
US dollars	9
Other dollars	10
Euros	12
Pounds	12
Negative currency	13
Thousands, millions, billions, and trillions	14
Floating point digits in currency expressions	14
Prenominal currency	15
Phone numbers	15
Social security numbers	16
Mixed alphanumeric tokens	16
Abbreviations	17
Uppercase acronyms and tokens	18
E-mail addresses	19
URLs	20
File names and paths	21

Punctuation.....	21
Parentheses.....	22
Hyphen.....	22
Slash.....	23
3. Embedded Tags	
Pronouncing numbers and years.....	25
English dates	26
Text locale tag.....	27
Pronouncing postal addresses.....	28
4. Symbolic Phonetic Representations	
Vowels.....	33
Consonants	34
Syllable stress.....	35
Syllable boundary.....	36
5. Phoneme marks	
6. User Dictionaries	
Main dictionary.....	41
Valid main dictionary entries.....	42
Main dictionary examples	42
Abbreviation dictionary	42
Valid abbreviation dictionary entries	43
Abbreviation dictionary examples.....	43
Root dictionary	43
Allowable root dictionary entries	44
Root dictionary examples	44
7. SSML	



Introduction

Overview

This document contains language-specific reference information for application developers using SpeechWorks Speechify 3.0 and higher.

Speechify™ is a Text-To-Speech (TTS) system.

New and changed information

Information about installing voices has been moved from this supplement to the *Speechify User's Guide*.

There are small improvements throughout this document.

See the *Speechify Migration Guide* for a complete list of changes in Speechify 3.0.

Getting started

To get started, we recommend that you are familiar with the *Speechify User's Guide*, which provides comprehensive installation, programming, and reference information about the Speechify product. You should also review the release notes distributed with this product for the latest information, restrictions, and known problems.

Other documents in the Speechify documentation set:

- ❑ *Speechify E-mail Pre-processor Developer's Guide* explains the types of input that the SpeechWorks E-mail Pre-processor handles, how it processes the messages, the modes that the application can take advantage of at runtime, the layout and use of the E-mail substitution dictionary, and provides the API function prototypes, types, error codes and constants.
- ❑ *Speechify Migration Guide* introduces several new features that have been added to Speechify 3.0 relative to Speechify 2.x releases.
- ❑ The *SpeechWorks Licensing Handbook* describes the process for getting licenses to run the Speechify software, details about configuring your license server, and related topics.
- ❑ There is a *Speechify Language Supplement* for each supported language.

Support services

To receive technical support from SpeechWorks International, Inc., use the following methods:

- ❑ Check the FAQs, visit the Knowledge Base, or ask a question at:

<http://techsupport.speechworks.com/>

- ❑ Ask for “technical support” at +1 617 428-4444



Standard Text Normalization

Speechify's text normalization is the component that interprets the input text and converts it into a sequence of fully spelled-out words. It encompasses digit conversion, abbreviation and acronym expansion, punctuation handling, and the interpretation of special characters such as currency symbols. It is this component of the system that is primarily responsible for an intelligent reading of text.

This chapter is a guide to the major features of Speechify's default text normalization, which applies unless overridden by a Main or Abbreviation Dictionary entry or an embedded tag. The *Speechify User's Guide* describes default normalization for all languages; below, the discussion describes behaviors specific to en-US.

The processing is described in general terms as an aid to application developers in constructing Main and Abbreviation Dictionaries, custom pre-processors, and marked-up text. It is beyond the scope of this chapter to provide an exhaustive description of the entire set of sophisticated, context-sensitive text normalization rules used in Speechify, and you may therefore observe deviations from the generalizations provided here in specific instances.

Definitions of symbols

Symbols used in the tables in this chapter are:

Symbol	Meaning
d or d (in recognized format column)	digit
a (in recognized format column)	lowercase letter
A (in recognized format column)	uppercase letter
<i>n</i> (in recognized format column)	a number
mm/dd/yy	month/day/year

Symbol	Meaning
hh:mm:ss	hour:minute:second
, (in text expansion examples)	pause and/or comma-like phrase break

Numbers

Cardinal numbers

Numbers are read as ordinary cardinal numbers by default. For example:

Example	Expansion
123	one hundred twenty three
25678	twenty five thousand six hundred
25,678	seventy eight

There are a few exceptions to this. Four-digit numbers with no thousands separator are pronounced as years by default. For example:

Example	Expansion
1200	twelve hundred
1984	nineteen eighty-four
2000	two thousand
2112	twenty-one twelve

Three-digit numbers are read in a shortened form, omitting the word “hundred,” in certain special contexts. For example:

Example	Expansion
I-494	ay four ninety four
Interstate 290	interstate two ninety

Four-digit numbers with no thousands separator are pronounced as years by default.

Example	Expansion
1200	twelve hundred
1984	nineteen eighty-four
2000	two thousand
2112	twenty-one twelve

Floating point digits

Recognized format	Example	Expansion
<i>n.n</i>	16.01	sixteen point zero one
<i>.n</i>	.345	point three four five

Negative numbers

Recognized format	Example	Expansion
<i>-n</i>	-37	minus thirty-seven

Fractions

Fraction characters are expanded as follows:

Fraction character	Example	Expansion
$\frac{1}{4}$	$\frac{1}{4}$	one fourth
	$3 \frac{1}{4}$	three and one fourth
$\frac{1}{2}$	$\frac{1}{2}$	one half
	$3 \frac{1}{2}$	three and a half

Fraction character	Example	Expansion
$\frac{3}{4}$	$\frac{3}{4}$	three fourths
	3 $\frac{3}{4}$	three and three fourths

Other fractions are ambiguous (e.g., $\frac{1}{2}$ can mean one half, one over two, one out of two, January first, and so on), and are interpreted as fractions only when occurring in a mixed number or in another disambiguating context.

Example	Expansion
2 $\frac{1}{2}$	two and a half
4 $\frac{1}{3}$	four and one third
He ate $\frac{3}{8}$ of an apple	he ate three eighths of an apple
$-\frac{5}{16}$	minus five sixteenths
$\frac{1}{4}$ km	one fourth kilometer
$\frac{2}{3}$ tsp salt	two thirds teaspoon salt

Zero-initial numbers

Numbers beginning with 0 are spoken digit by digit.

Example	Expansion
02319	zero two three one nine

Numeric expressions

Dates

The following table shows the formats that Speechify recognizes as dates. In the table, parentheses enclose optional elements, and *mon* designates a three-letter month abbreviation.

Recognized format	Example
(m)m/(d)d/(yy)yy	1/10/1984
(m)m-(d)d-(yy)yy	02-23-86
(d)d/(m)m/(yy)yy	10/25/88
(d)d-(m)m-(yy)yy	5-5-1991
(d)d-mon-(yy)yy	25-Aug-01
yyyy-mon-(d)d	2002-Apr-03

Many dates in English are ambiguous between the *mdy* (month-day-year) format used in the US, and the *dmy* (day-month-year) format used in Great Britain, Australia, and elsewhere. For example, the date 1/10/1984 is normally read as January tenth, 1984, in the US, and as October first, 1984, in Australia and Great Britain. This ambiguity exists regardless of which Speechify voice is reading a text, because a voice can be used to read a text from a locale other than its native locale.

Speechify produces a fully expanded date reading of dates that are not ambiguous. This includes a full expansion for the month and an ordinal reading of the day. For example:

Input	Expansion
12/31/1956	december thirty-first, nineteen fifty-six
15/04/2001	april fifteenth, two thousand one
5/5/1991	may fifth, nineteen ninety-one
2002-Apr-01	april first, two thousand two

In addition, when a date is fully expanded, a two-digit year is interpreted according to the following rules:

Two-digit year	Interpretation
00–09	2000–2009
10–29	10–29 (neutral as to century)
30–99	1930–1999

For example:

Input	Expansion
2/23/86	february twenty-third, nineteen eighty-six
5/17/03	may seventeenth, two thousand three
25-Aug-01	august twenty-fifth, two thousand one

Speechify also gives a fully expanded date reading when a date or text locale tag has been used to specify date format (see “English dates” on [page 26](#) and “Text locale tag” on [page 27](#)).

In all other cases, Speechify reads dates numerically, to prevent an inaccurate rendering. In this case, two-digit years are read neutrally as to century. For example:

Input	Expansion
3/4/02	three four oh two
9/5/1956	nine five nineteen fifty-six

Days of the month

Days of the month are converted to ordinal numbers. For example:

Input	Expansion
October 25, 1988	october twenty-fifth, nineteen ninety-eight
Friday, Jan. 10	friday, january tenth
Dec. 22-25	december twenty-second to twenty-fifth
August 1, 2, and 3	august first, second, and third
23 Feb 1986	february twenty-third nineteen eighty-six
He arrived on May 5.	he arrived on may fifth.

Times

In the following table, parentheses enclose optional elements. Note that times delimited by periods are interpreted as times only if an am/pm marker follows. Seconds, if included, are not pronounced.

Recognized format	Example	Expansion
(h)h:mm	12:01	twelve oh one
	14:52	fourteen fifty-two
	9:00	nine o'clock
	08:35	eight thirty-five
(h)h:mm am/pm marker (see next column for am/pm marker formats)	10:30am	ten thirty ay emm
	10:30AM	
	10:30 am	
	10:30 AM	
	10:30 a.m.	
	10:30 A.M.	
(h)h:mm am/pm marker (see next column for am/pm marker formats)	10.00pm	ten pea emm
	10.00PM	
	10.00 pm	
	10.00 PM	
	10.00 p.m.	
	10.00 P.M.	
(h)h:mm:ss	8:23:34	eight twenty-three
(h)h:mm.ss am/pm marker	9.14.14 pm	nine fourteen pea emm

Currency

US dollars

Dollar expressions with no explicit nationality indicator are read neutrally as dollars. If US dollars are specified explicitly, they are read as such.

Recognized format	Example	Expansion
<i>\$n</i>	\$100	one hundred dollars
<i>n\$</i>	100\$	
<i>\$USn</i>	\$US100	one hundred you ess dollars
<i>US\$n</i>	US\$100	

Recognized format	Example	Expansion
$\$n$ US	\$100US	
n USD	100USD	
n USD	100 USD	
n (US\$)	100 (US\$)	

Any of the formats above may include cents. Examples:

Example	Expansion
\$0.50	fifty cents
\$.50	fifty cents
\$5.10	five dollars and ten cents
\$US5.10	five you ess dollars and ten cents
US\$5.10	five you ess dollars and ten cents
\$5.10US	five you ess dollars and ten cents
5.10USD	five you ess dollars and ten cents
5.10 USD	five you ess dollars and ten cents
5.10 (US\$)	five you ess dollars and ten cents

Other dollars

Nationality	Recognized format
Australian dollars	$\$A$ n
	$\$AU$ n
	$\$AUS$ n
	A $\$$ n
	AU $\$$ n
	AUS $\$$ n
	$\$n$ AU
	$\$n$ AUS
	$\$n$ AU
	$\$n$ AUS
	n AUD
	n AUD

Nationality	Recognized format
Canadian dollars	$\$Cn$
	$\$CANn$
	$\$CADn$
	$\$CDNn$
	$C\$n$
	$CAN\$n$
	$CAD\$n$
	$CDN\$n$
	$\$nCAN$
	$\$nCAD$
	$\$nCDN$
	$\$n\ CAN$
	$\$n\ CAD$
	$\$n\ CDN$
	$nCDN$
	$n\ CDN$
	$n\ (CAN\$)$
	$n\ (CAD\$)$
	$n\ (CDN\$)$
Hong Kong dollars	$\$HKn$
	$HK\$n$
	$\$nHK$
	$\$n\ HK$
	$nHKD$
	$n\ HKD$
Singapore dollars	$\$S\n
	$S\$n$
	$n\$GD$
	$n\ \$GD$
New Zealand dollars	$\$NZn$
	$NZ\$n$
	$\$NZD\ n$
	$NZD\$n$
	$n\ NZD$
	$\$n\ NZD$

All of the formats shown above may include subunits, read as cents. A few examples:

Example	Expansion
\$AUS25.39	twenty-five australian dollars and thirty-nine cents
A\$25.39	
\$11.20CDN	eleven canadian dollars and twenty cents
CAN\$11.20	

Euros

Recognized format	Example	Expansion
EUR <i>n</i>	EUR 100	one hundred euros
EUR <i>n</i>	EUR100	
<i>n</i> EUR	100 EUR	
<i>n</i> EUR	100EUR	
€ <i>n</i>	€ 100	
€ <i>n</i>	€100	
<i>n</i> €	100 €	
<i>n</i> €	100€	

The euro formats shown above may include subunits (cents). Examples:

Example	Expansion
EUR 5.10	five euros and ten cents
EUR5.10	
5.10 EUR	
5.10EUR	
€ 5.10	
€5.10	

Pounds

Recognized format	Example	Expansion
£ <i>n</i>	£ 100	one hundred pounds
£ <i>n</i>	£100	

Recognized format	Example	Expansion
£STG <i>n</i>	£STG 100	one hundred pounds sterling
£STG <i>n</i>	£STG100	
£Stg <i>n</i>	£Stg 100	
£Stg <i>n</i>	£Stg100	
£stg <i>n</i>	£stg 100	
£stg <i>n</i>	£stg100	

The pound formats shown above may include subunits (pence). Examples:

Example	Expansion
£ 5.10	five pounds and ten pence
£5.10	five pounds and ten pence
£STG 5.10	five pounds sterling and ten pence
£STG5.10	five pounds sterling and ten pence
£Stg 5.10	five pounds sterling and ten pence
£Stg5.10	five pounds sterling and ten pence
£stg 5.10	five pounds sterling and ten pence
£stg5.10	five pounds sterling and ten pence

Negative currency

Negative currency values can be denoted with a minus sign immediately preceding the number. For example:

Example	Expansion
\$-10.00	minus ten dollars
-100HKD	minus one hundred hong kong dollars
£Stg -34	minus thirty-four pounds sterling

In dollar expressions, the minus sign may also precede the dollar sign. For example:

Example	Expansion
-\$10	minus ten dollars
-\$5.50	minus five dollars and fifty cents
-\$3US	minus three you ess dollars

Thousands, millions, billions, and trillions

Thousands, millions, billions, and trillions may be explicitly denoted with the appropriate word or abbreviation. Full words may be all lowercase, all uppercase, or capitalized. The following table shows dollars, but any of the supported currency types and formats is possible. The number *n* may include floating point digits.

Recognized format	Example	Expansion
$\$n$ thousand	\$10 thousand	ten thousand dollars
$\$n$ -thousand	\$4.5-thousand prize	four point five thousand dollar prize
$\$n$ million	\$5.5 million	five point five million dollars
$\$nM$	\$5.5M	
$\$mn$	\$5.5m	
$\$mn$	\$5.5mn	
$\$n$ -million	\$4-million-a-year profit	four million dollar a year profit
$\$n$ billion	\$1.039 billion	one point zero three nine billion
$\$nb$	\$1.039b	
$\$nb$	\$1.039bn	
$\$n$ -billion	\$1-billion goal	one billion dollar goal
$\$n$ trillion	\$17.67 trillion	seventeen point six seven trillion dollars
$\$ntn$	\$17.67tn	
$\$n$ -trillion	\$3.2-trillion debt	three point two trillion dollar debt

Floating point digits in currency expressions

Any currency type may have floating point digits rather than subunits. For example:

Example	Expansion
\$100.345	one hundred point three four five dollars
£1.1	ten point one pounds
\$A5.5	five point five australian dollars

Prenominal currency

When it can be determined with confidence that a currency expression modifies a following noun, the currency word is singular.

Example	Expansion
Tom has \$500 in the bank.	tom has five hundred dollars in the bank.
Tom won a \$500 prize.	tom won a five hundred dollar prize.

Phone numbers

Recognized phone numbers are read out digit by digit, with pauses in the locations indicated by commas in the Expansion column below. Note that the words area code are prefixed to the area code only when it is enclosed in parentheses, since otherwise the digit sequence may not be a phone number.

Recognized format	Example	Expansion
ddd-dddd	234-8390	two three four, eight three nine zero
ddd-ddd-dddd	617-794-8390	six one seven, seven nine four, eight three nine zero
ddd ddd-dddd	617 794-8390	
ddd/ddd-dddd	617/794-8390	
(ddd) ddd-dddd	(617) 794-8390	area code six one seven, seven one four, eight three nine zero
(ddd)ddd-dddd	(617)794-8390	
1-ddd-ddd-dddd	1-800-555-2076	one, eight hundred, five five five, two zero seven six
1-ddd-AAA-AAAA	1-800-YOU-PICK	one, eight hundred, you pick
1.ddd.ddd.ddd	1.617.794.8390	one, six one seven, seven nine four, eight three nine zero

A phone number may be followed by an extension. The phone number and extension are separated by a pause.

Recognized format	Example	Expansion
ext. <i>n</i>	ext. 34	extension thirty-four
ext. <i>n</i>	ext.34	
ext <i>n</i>	ext 34	
ex. <i>n</i>	ex. 34	
ex. <i>n</i>	ex.34	
ex <i>n</i>	ex 34	
xt. <i>n</i>	xt. 34	
xt. <i>n</i>	xt.34	
xt <i>n</i>	xt 34	
x. <i>n</i>	x. 34	
x. <i>n</i>	x.34	
x <i>n</i>	x 34	
x <i>n</i>	x34	

Social security numbers

Format	Example	Expansion
ddd-dd-dddd	625-17-9034	six two five, one seven, nine zero three four

Mixed alphanumeric tokens

Mixed alphanumeric sequences are either spelled, divided into words, or in special cases read as whole words. For example:

Example	Expansion
55th	fifty-fifth
1930s	nineteen thirties
3RD	third
VOS34	vee oh ess thirty-four
32XC	thirty-two eks cee

Example	Expansion
J2EE	jay two ee ee
Group12	group twelve
24hour	twenty-four hour
12min	twelve minutes

Abbreviations

Ambiguity in abbreviations

There are two types of ambiguity in abbreviations. The first type of ambiguous abbreviation has more than one possible expansion; e.g., St. stands for either street or saint. The second type of ambiguous abbreviation can be either a whole word or an abbreviated form: for example, “in.” is either the word “in” or the abbreviation for “inches.” Speechify employs a set of context-sensitive rules to disambiguate the first type of ambiguous abbreviation.

A token of the second type is interpreted as an abbreviation only in specific disambiguating contexts. For example, when the abbreviation ends in a period, and the following word begins with a lowercase letter, it is interpreted as an abbreviation. For example:

Input	Expansion
There are 5 in. of snow on the ground.	there are five inches of snow on the ground.
Put 5 in. Then take 3 out.	put five in. then take three out.

Periods in abbreviations

Note that abbreviations do not necessarily end in a period:

Input	Expansion
We met Mr Smith yesterday.	we met mister smith yesterday.

Once the abbreviation is expanded, if it ended in a final period it must be determined whether or not the period also indicates a sentence end:

Input	Expansion
There are 5 ft. of snow on the ground.	there are five feet of snow on the ground.
It snowed 5 ft. The next day it all melted.	it snowed five feet. the next day it all melted.

Measure abbreviations are expanded to singular or plural depending on the value of a preceding number or other disambiguating context. In the absence of a disambiguating context, they default to a plural expansion. For example:

Input	Expansion
There are 5 in. of snow on the ground.	there are five inches of snow on the ground.
There is 1 in. of snow on the ground.	there is one inch of snow on the ground.
There are 1.1 in. of snow on the ground.	there are one point one inches of snow on the ground.
How many cm. are in a km?	how many centimeters are in a kilometer?
cm.	centimeters

Uppercase acronyms and tokens

Uppercase acronyms and other tokens written in uppercase are either read out letter by letter, read as a whole word, or given an idiosyncratic interpretation. Examples:

Example	Pronunciation
TTS	tee tee ess
CMU	cee emm you
CIA	cee eye ay
AAA	triple ay
UNICEF	unicef
NYSE	new york stock exchange

A possessive suffix can be attached to an acronym without interfering with the ordinary acronym processing. For example:

Input	Expansion
CIA's	cee eye ay's
UNICEF's	unicef's

E-mail addresses

An e-mail address is divided into two portions, a user name and a domain name, separated by an @. A phrase break is inserted following the user name. Symbols in the e-mail address are read as follows:

Symbol	Expansion
@	at
.	dot
-	dash
_	underscore

User name

The user name is spelled out character by character, unless word boundaries are indicated unambiguously. Sequences of two and three letters are always spelled out. Digit sequences are read digit by digit. Examples:

User name	Expansion
bruce_smith	bruce underscore smith
john.jones	john dot jones
star-chaser	star dash chaser
steve5050	steve five zero five zero
red	ar ee dee
rfrmac	ar eff ar emm ay cee

Domain name

Two- and three-letter domain and country extensions are either read as words or spelled out, following standard convention. The host name is read as a single word, unless word boundaries are indicated unambiguously. Examples:

Host name	Expansion
access1.net	access one dot net
hawaii.rr.com	hawaii dot ar ar dot com
cornell.edu	cornell dot ee dee you
amazon.co.uk	amazon dot cee oh dot you kay

URLs

A token beginning with `www.` or `http://` or `ftp://` is interpreted as a URL. A phrase break is inserted following `http://` or `ftp://`, and the `://` are not pronounced.

Symbols in a URL are expanded as follows:

Symbol	Expansion
/	URL-final: not expanded (otherwise: slash)
.	dot
-	dash
_	underscore

Each slash-delimited segment of the URL is expanded as follows: Two- and three-letter domain and country extensions are either read as words or spelled out, following standard conventions. Each remaining segment is read as a single word, unless word boundaries are indicated unambiguously. Examples:

URL	Expansion
<code>http://www.lobin.freemove.co.uk/sarahs-price/page001.html</code>	aitch tee tee pee, double you double you double you dot lobin dot freemove dot cee oh dot you kay slash sarahs dash price slash page zero zero one dot aitch tee em ell
<code>www.serbia-info.com/news/</code>	double you double you double you dot serbia dash info dot com slash news

File names and paths

Symbols in paths are expanded as follows:

Symbol	Expansion
/	URL-final: not expanded (otherwise: slash)
\	URL-final: not expanded (otherwise: backslash)
:	colon
.	dot
-	dash
_	underscore

Each slash-delimited segment of a pathname is read as a single word, unless word boundaries are unambiguously indicated. Common filename extensions are read as a word or spelled out, following standard conventions. Examples:

Pathname	Expansion
C:\docs\my_book\chapter12.doc	cee colon backslash docs backslash my underscore book backslash chapter twelve dot doc
/product/release/speechify-3-0-0/release-notes.txt	slash product slash release slash speechify dash three dash oh dash oh slash release dash notes dot tee eks tee

Punctuation

Punctuation generally triggers a phrase break except in a limited set of special cases that are determined on a language-specific basis. Examples of these are:

Input	Expansion
He lives in Boston, Massachusetts.	he lives in boston massachusetts.
That is a very, very old building.	that is a very very old building.
John won't come, either.	john won't come either.
SpeechWorks International, Inc.	speechworks international incorporated.
Drive D: is full.	drive dee is full.

Parentheses

Parentheses generally trigger a phrase break:

Input	Expansion
Tom (my son) and Susan (my daughter)	Tom, my son, and Susan, my daughter

The phrase break does not happen in a very limited set of special cases.

Input	Expansion
book(s)	books
getText()	get text

Hyphen

The hyphen is read neutrally as “dash” unless it can be disambiguated with a high degree of confidence. Examples:

Input	Expansion
mid-90s	mid nineties
A-1	ay one
32-bit	thirty-two bit
-7	minus seven
April 3-4	April third to fourth
pp. 35-40	pages thirty-five to forty
1974-1975	nineteen seventy-four to nineteen seventy-five
2-3 inches	two to three inches
3-2	three dash two (because it could be “3 to 2”; “3 minus 2”; or “3 2”)

Slash

A slash is read as “slash” unless the following word is a unit of measure, when the slash is read as “per,” or when the entire token is a familiar expression.

Input	Expansion
Seattle/Boston	seattle slash boston
cm/sec	centimeters per second
he/she	he or she



Embedded Tags

Embedded tags are special codes that can be inserted into input text to customize Speechify's behavior in a variety of ways. The *Speechify User's Guide* describes the Speechify tag format and functionality. Below is the description of a tag that is specific to US English.

Pronouncing numbers and years

In US English, a four digit numeric sequence with no internal commas or trailing decimal digits, like *1984*, can be interpreted either as a year (*nineteen eighty four*) or as a quantity (*one thousand nine hundred eighty four*). Speechify applies the year interpretation by default, as in:

Input	Pronunciation
He was born in May 1945.	He was born in May nineteen forty five.

To override or restore the default year interpretation, use the following tags:

Year mode tag	Description
\ny0	Quantity interpretation.
\ny1	Year interpretation (default).

For example:

Input	Pronunciation
In May \ny0 1945 people emigrated.	In May one thousand nine hundred forty five people emigrated.

Each tag remains in effect until the interpretation is toggled by the use of the other tag (or until the end of the speak request). For example:

Input	Pronunciation
\ny0 1945 \ny1 people emigrated in 1945.	One thousand nine hundred forty five people emigrated in nineteen forty five.

English dates

Use the date tag to specify the format of an immediately following date. This tag allows you to specify the order of the month and day elements in a date, so that month names can replace the neutral numeric reading.

Date tag	Description
\date:dmy	The following text is a date in day, month, year order.
\date:mdy	The following text is a date in month, day, year order.

For example:

Input	Pronunciation
\date:dmy 3/4/1998	april third, nineteen ninety-eight
\date:mdy 3/4/1998	march fourth, nineteen ninety-eight

**NOTE**

- ❑ The date tag applies only to the immediately following token. If that token is not a date, the tag is ignored.
- ❑ The date tag works correctly only if the following date is in a recognized date format. (See “Dates” on [page 7](#) for details on recognized formats.) The date tag cannot be used to force a date reading of other date formats.
- ❑ You can specify the default interpretation of dates within an entire document with the text locale tag.
- ❑ The date tag overrides a default specification from a text locale tag for the immediately following token.
- ❑ If the format specified in a date tag is inconsistent with the contents of the date, the tag is ignored. In this example, the “21” cannot refer to a month:

Input	Pronunciation
<code>\!date:dmy 4/21/1958</code>	april fourth, nineteen fifty-eight

Text locale tag

Use the locale tag at the beginning of a document to specify the locale of that document’s origin. This tag currently controls default date formats.

Locale tag	Description
<code>\!locale:US</code>	Assume all dates are mdy format.
<code>\!locale:GB</code>	Assume all dates are dmy format.
<code>\!locale:AU</code>	Assume all dates are dmy format.

Note that the date tag can override the locale tag for a single date. The default setting resumes following that date. For example:

Input	Pronunciation
<code>\locale:US 3/4/1998,</code> <code>\date:dmy 5/6/1984, and</code> <code>7/8/1981</code>	march fourth, nineteen ninety-eight, june fifth, nineteen eighty-four, and july eighth, nineteen eighty-one

Pronouncing postal addresses

Use “address mode” to improve the interpretation of postal addresses for the United States and English-speaking Canada. The address tags enclose a single address or a list of addresses:

Tag	Description
<code>\addr1</code>	Begin address mode.
<code>\addr0</code>	End address mode.

Address mode remains in effect until turned off with the end tag (or until the end of the speak request).

Address formats

Addresses have the following parts. *Separate each part using commas or newlines.*

- ❑ Street address — a street number and name with typical abbreviations. Optionally includes a secondary unit such as apartment, suite, floor, or building number.
- ❑ Post office box number — can appear before, after, or in place of a street address.
- ❑ City-State — can include a postal code and country name.

Use address delimiters to improve the speech

To improve the resulting speech, insert newline or comma delimiters between address parts. Otherwise, any ambiguities will degrade the spoken output. For example:

```
5 Smith's Way S. Dartmouth MA 02748
```

How should the address above be read? The boundary between street address and city-state cannot be determined. Below, commas are inserted for clarification:

```
5 Smith's Way S., Dartmouth MA 02748 OR
5 Smith's Way, S. Dartmouth MA 02748
```

In the first line of the next example, should “St.” be Street or Saint? In the second line, it is clearly “Saint;” in the third, it is “Street.”

```
6800 Sunshine Skyway St. Petersburg FL 33711
6800 Sunshine Skyway, St. Petersburg, FL 33711 OR
6800 Sunshine Skyway St., Petersburg, FL 33711
```

Optionally, you can use a newline or comma to separate a secondary unit (such as apartment, suite, floor, or building number) from the main part of the street address. The following examples are pronounced in the same way:

```
1919 Clement Ave. Bldg. 11
```

```
1919 Clement Ave., Bldg. 11
```

```
1919 Clement Ave.
Bldg. 11
```

In a list of addresses (where address mode is turned ON for more than one address), each address should end with a double newline. Alternatively, you can enclose each address separately within address tags. That is, the following are equivalent:

```
\!addr1
600 E. Superior St., Duluth, MN 55802
```

```
11 Hackett Blvd., Albany, NY 12208
\!addr0
```

```
\!addr1600 E. Superior St., Duluth, MN 55802 \!addr0
\!addr1 11 Hackett Blvd., Albany, NY 12208 \!addr0
```

Examples of addresses

Comma-delimited:

9744 Wilshire Blvd., Ste. 307, Beverly Hills, CA 90212

P.O. Box 400, 540 Thompson Creek Rd., Stevensville, MD 21666

Newline-delimited:

5324 Calgary Trail N, Ste. 200
Edmonton, AB T6H 4J7

3162 Matecumbe Key Rd.
Punta Gorda, FL 33955

1501 Powell St., Suite F
Emeryville, CA 94608

P.O. Box 925
8 Heritage Park Rd.
Clinton, CT 06413

91 Cayou Quay
P.O. Box 55
Deer Harbor, WA 98243-0055

Avoid extraneous text

Once address mode begins, all text is assumed to be part of the address, so you should avoid including non-address text inside the tags. A contrived example:

```
\!addr1 S. Neville said 34567 Ave Marias. \!addr0
```

The string above is read “South Neville said three four five six seven Avenue Marias.”

However, addresses may include a limited amount of additional text such as company or building names. For example:

The Greenstate Building
127 W. State St.
Ithaca, NY 14850

How numbers are processed

The following rules apply to cardinal numbers in addresses, such as house numbers, apartment numbers, and postal codes. Ordinal numbers within addresses are read as usual. (For example, *3rd St* is read “Third Street.”)

- ❑ 1–2 digits — Read as ordinary numbers.
- ❑ 3 digits
 - If the digits end in zeros (x00), the number is read “x hundred”
 - If the middle digit is zero (x0y), the number is read “x oh y”
 - If there are no zeros (xyz), the number is read “x yz” (for example, 367 is read as “three sixty-seven”)
- ❑ 4 digits:
 - If the digits end in zeros (x000), the number is read “x thousand”
 - If the third digit is zero (xy0z), the number is read “xy oh z” (for example, 2305 is read “twenty-three oh five” and 2005 is read “twenty oh five”)
 - Otherwise, for (xyzw) the number is read “xy zw” (for example, 2345 is read “twenty-three forty-five”)
- ❑ 5 digits or more — Read digit by digit.

Abbreviation processing

Address mode handles abbreviations as follows:

- ❑ Directions are expanded; for example N., S.W., and SE, become “North,” “Southwest,” and “Southeast.”
- ❑ Two-letter abbreviations for US states and Canadian provinces are generally expanded whether they are written in uppercase letters, lowercase letters, or mixed case. For example, “MN,” “Mn,” and “mn” are all expanded as “Minnesota” in the following city-state lines:

```
Duluth, MN 55802
Duluth, Mn 55802
Duluth, mn 55802
```

- ❑ Expansion is sensitive to position within the address. For example:

```
12 Auburn Ct
Haverford, Ct

245 Hole in the Bucket Lane
muncie, in 47305
```

In the first address above, “Ct” is “Court” in the street address part and “Connecticut” in the city-state part. In the second address, “in” is read as “Indiana” in the city-state part.

Prosody

Address mode inserts phrase boundaries and accompanying pauses as follows:

- ❑ Between the street address and the post office box, if any
- ❑ Between the street address and the secondary designator (e.g. apartment number), if any
- ❑ Before the city-state part
- ❑ Between the state and the postal code

Address mode also inserts a sentence boundary at the end of each address when the address is delimited by a double newline. No sentence boundaries are inserted within an address.

Address fragments

Addresses do not need to be complete. The input text can contain address fragments (for example, either the street address part or the city-state part).



Symbolic Phonetic Representations

The following tables show the inventory of available symbols for use in US English Symbolic Phonetic Representations (SPRs). Use this chapter with the SPR information found in the *Speechify User's Guide*.

Each sound symbol is accompanied by examples illustrating typical spellings of the sound in actual words, with bold letters representing the given sound. Due to dialectal differences, the SPR examples shown may not always agree with your own pronunciations.

Vowels

The following table includes the US English symbols for vowels.

US English symbol	Example words
i	see , speak , believe
I	pick , ill
e	cake , pain
E	hedge , let
A	back , had
H	mug , son
R	butter , hurt
u	zoo , truth
U	took , put

US English symbol	Example words
o	both, oak
c	law , cough
a	father, lot
x	sofa, alone
X	roses, hinted
Y	life, fly, bonsai, Einstein
O	toil, boy
W	out , cow

Consonants

The following table includes the US English symbols for consonants.

US English symbol	Example words
p	pit , rip
b	bad , sob
t	tip, pet
d	dip , had
k	kill , cat, make, back
g	good, bug
C	chip , witch, nature
J	Jane , huge
f	field, if, graph
v	vase, save
T	thing , Beth
D	this , breathe
s	seal, miss, ceiling
z	zip, phase

US English symbol	Example words
S	ship , wish
Z	treasure, garage
h	hot , hero
m	man , hum , summer
n	never , sun , winner
N (“syllabic n”)	button , satin , burden
G	sing , finger
l	low , hall
r	borrow , rake
F (“flap”)	writer , fiddle
w	wear , quick
y	yes , Virginia

Syllable stress

You can mark syllables for stress with a digit. The following table includes the US English symbols for syllable stress.

0	no stress
1	primary stress (most prominent stress in the word)
2	secondary stress

If a word has more than one syllable, you can mark one of these syllables for primary stress, and mark others for secondary stress or no stress.

A syllable that is not marked for stress is assumed to have no stress, unless it is the only syllable of a word, in which case it is assigned a primary stress.

The syllable stress marker should be within the boundaries of the syllable, and to the left of the vowel. If you do not know where the syllable boundaries in a word like *construction* are located, any of the following SPRs correctly place the primary stress on the second vowel:

```
construction
\! [kXn1strHkSXn]
\! [kXns1trHkSXn]
\! [kXnst1rHkSXn]
\! [kXnstr1HkSXn]
```

Syllable boundary

The following table includes the US English symbol for a syllable boundary.

.	beginning of a syllable
---	-------------------------



Phoneme marks

The phoneme-mark data structure is useful for matching each moment of synthesized speech with the phonemes being spoken. This information is particularly useful when you synchronize facial animation with speech output.

The structure is returned to your application's callback function, which is described under “SWIttsCallback()” in the API reference section of the *Speechify User's Guide*. Each phoneme mark contains the phoneme name, along with other information about the phoneme. The phoneme-mark symbols for US English are documented in the following table.

Note: The phoneme names used in phoneme marks are not necessarily the same as those used for SPR input.

Phone	Example
aa	rod , father
ae	back, had
ah	but, mug, son
ao	law , cough
aw	out , cow
ax	sofa, alone
ay	life, fine
b	bad , sob
ch	chip , witch
d	dip , had

Phone	Example
dh	this , breathe
dx	writer, bet ting
eh	hedge, le t
en	button , e aten
er	butter , h urt
ey	cake, pain [a, ai]
f	fi eld, if , graph h
g	g ood, bug
hh	h ot, h ero
ih	pi ck, i ll
iy	see , spea k, believe
jh	J ane, hug e
k	k ill, make
l	l ow, h all
m	man , h um, summer
n	n ever, sun , wi n
ng	si ng, fi nger
ow	bo th, o ak
oy	to il, bo y
p	pi t, ri p
pau	[Indicates a pause]
r	bor ro w, ra ke
s	se al, mi ss, ce iling
sh	sh ip, wi sh
t	ti p, pe t, ki tten
th	th ing, B eth
uh	to ok, pu t

Phone	Example
uw	zoo , truth
v	v ase, sav e
w	w ear, qu ick
y	y es, Virgini a
z	z ip, phas e
zh	treasur e, garag e



User Dictionaries

This chapter describes US English behaviors for Speechify dictionaries. Use this chapter in conjunction with the dictionary information found in the *Speechify User's Guide*.

The maximum length for US English dictionary keys is 128 characters. The maximum translation length is 512 characters.

Main dictionary

The main dictionary is an all-purpose user dictionary for replacing a word in an input text with almost any type of input string. Main dictionary entries are case-sensitive.

For US English, lookups in the main dictionary ignore an attached apostrophe followed by the letter *s* (*'s*) as used to indicate possession. For example, if *WHO* is a main dictionary key translating to *World Health Organization*, it matches an input string *WHO's offices*, yielding the pronunciation *World Health Organization's offices*.

Valid main dictionary entries

The following table summarizes the valid main dictionary keys and translations:

Valid key requirement	Valid translation
Latin 1 letters (both upper and lower case), digits. Non-alphanumeric characters, including: @, #, \$, %, &, *, + Apostrophes, quotation marks, parentheses, brackets, etc. Punctuation, except as the final character.	Anything that is valid input to the text-to-speech engine, including white space, punctuation, SPRs, and embedded tags.
Disallowed: white space	Disallowed: SAPI tags, SSML tags, and bookmarks

Main dictionary examples

The following table shows examples of main dictionary entries:

Key	Translation
AWSA	American Woman Suffrage Association
stone@ccom.org	ess tee one at c com dot o r g
UConn	\![2yu1kan]
WYSIWYG	\![1wIziwIg]
486DX	4 eighty six \![di] ecks

Abbreviation dictionary

The abbreviation dictionary handles word abbreviations that translate to one or more words in ordinary spelling. The entries are case-sensitive.

For US English, lookups in the abbreviation dictionary ignore an attached apostrophe followed by the letter s (s) as used to indicate possession. For example, if *Inc.* is an abbreviation dictionary key translating to *Incorporated*, it matches an input string *SpeechWorks International Inc.'s offices*, yielding the pronunciation *SpeechWorks International Incorporated's offices*.

Valid abbreviation dictionary entries

The following table summarizes the format of valid keys and translations in the abbreviation dictionary:

Valid key requirement	Valid translation
Sequences of one or more letters optionally separated by periods (x.x.x or xx.xx.xx). Sequences of letters, with or without the trailing period that may be considered part of the abbreviation (xxx. or xxx). Internal apostrophes (not the first or last character in the sequence). Uppercase or lowercase letters.	One or more valid words in ordinary spelling, including both upper and lower case letters, separated by white space or hyphen.
Disallowed: digits, non-alphabetic symbols, white space, or punctuation (except periods).	Disallowed: digits, punctuation, SPRs, or embedded tags.

Abbreviation dictionary examples

The following table shows examples of abbreviation dictionary entries:

Key	Translation
Is.D.	eye ess dee
Ltjg	Lieutenant Junior-Grade
int'l	international

Root dictionary

The root dictionary is used for ordinary words, like nouns (including proper names), verbs, or adjectives. Unlike the main and abbreviation dictionaries, it is not case-sensitive.

The US English root dictionary requires that all forms of a word (for example, singular and plural forms) must be entered separately. You must create separate root dictionary entries for each word that is based on a given root in order to customize

the pronunciation in all cases. For example, you should create individual root dictionary entries for *roof*, *roofs*, *roof's*, *roofed*, *roofing*, *roofer*, *reroof*, etc., in order to generate the desired pronunciation of each word.

Allowable root dictionary entries

The following table summarizes the format of valid keys and translations in the root dictionary:

Key	Translation
A single word in ordinary spelling, all lowercase letters.	A single word in ordinary spelling. A valid SPR.
Disallowed: digits, punctuation, or other non-alphabetic characters; white space	Disallowed: digits, punctuation, or other non-alphabetic characters, white space, tags, and annotations

Root dictionary examples

The following table shows examples of root dictionary entries:

Key	Translation	Would apply to...
figure	\[.1fl.0gR]	figure, Figure
tomato	\[.0tx.1ma.0to]	tomato, Tomato
wilhelmina	wilma	wilhelmina, Wilhelmina



SSML

Speechify includes support for input text formatted according to the Speech Synthesis Markup Language (SSML).

In order to invoke the correct processing for US English text, the given SSML input must have the `xml:lang` attribute set to “en-US.” If this attribute is absent, Speechify assumes “en-US” as a default.

In addition to producing US English output, the SSML processor for en-US parses and expands tag content in a manner consistent with that locale. For example, the tag `<say-as type = "currency">10.20</say-as>` is pronounced “ten dollars and twenty cents.”

For more description of SSML support in Speechify, see the *Speechify User's Guide*.

Index

Symbols

\!addr0 28

\!addr1 28

\!date 26

\!locale 27

\!ny0 25

\!ny1 25

A

abbreviation dictionary 42

abbreviations 17

acronyms 18

address mode 28

alphanumeric sequences 16

area code 15

Australian dollars 10

B

billions 14

C

Canadian dollars 11

cardinal numbers 4

consonants 34

currency 9

 Australian dollars 10

 Canadian dollars 11

 euros 12

 floating point numbers 14

 Hong Kong dollars 11

 New Zealand dollars 11

 other dollars 10

 pence 13

 pounds 12

 prenominal 15

 Singapore dollars 11

 US dollars 9

D

dash 22

date embedded tag 26

dates 7

dictionaries

 user 41

dictionary

 abbreviation 42

 main 41

 root 43

E

e-mail addresses 19

embedded tags 25

 date 26

 dates 26

 locale 27

 postal addresses 28

 pronouncing numbers and years 25

English SPRs 33

euros 12

extension 16

F

file names 21

floating point numbers 5

 currency 14

fractions 5

H

Hong Kong dollars 11

hyphen 22

L

locale embedded tag 27

M

main dictionary 41

millions 14

N

negative currency 13

negative numbers 5

New Zealand dollars 11

numbers

 billions 14

 cardinal 4

 floating point 5

 fractions 5

 millions 14

 negative 5

 thousands 14

 trillions 14

 zero-initial 6

numbers and years

 embedded tag 25

numeric expressions 7

P

parentheses 22

paths 21

pence 13

phone numbers 15

phoneme mark 37

postal addresses

 embedded tags 28

pounds 12

prenominal currency 15

punctuation [21](#)

R

root dictionary [43](#)

S

Singapore dollars [11](#)

slash [23](#)

social security numbers [16](#)

SPR symbols

consonants [34](#)

US English [33](#)

vowels [33](#)

stress, syllable [35](#)

support services [2](#)

syllable boundary [36](#)

syllable stress [35](#)

T

thousands [14](#)

times [9](#)

trillions [14](#)

U

URLs [20](#)

US dollars [9](#)

user dictionaries [41](#)

V

vowels [33](#)

Z

zero-initial numbers [6](#)