Logging Variables

General variables

%YCurrent year as YYYY%yCurrent year as YY%MCurrent month as MM%DCurrent day as DD%hCurrent hour as HH%mCurrent minute as MM%sCurrent second as SS%wIOS8601 week of the year%If Format JCurrent date/time, using FormatDateTime formatting (see below)%s{Format JStart time in FormatDateTime formatting (used for stop logging)%rCa character (ASCII 9)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%gKuntime data%#Total number of current listeners on all encoder connections		
Add to the second as SM%MCurrent month as MM%DCurrent day as DD%hCurrent hour as HH%mCurrent minute as MM%sCurrent second as SS%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%Ү	Current year as YYYY
%DCurrent day as DD%hCurrent hour as HH%mCurrent minute as MM%sCurrent second as SS%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%gUnique identifier of the associated playback operation%g{Key}Runtime data	%у	Current year as YY
%hCurrent hour as HH%mCurrent minute as MM%sCurrent second as SS%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%M	Current month as MM
%mCurrent minute as MM%sCurrent second as SS%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%D	Current day as DD
%sCurrent second as SS%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%h	Current hour as HH
NumberInstantion%wIOS8601 week of the year%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%m	Current minute as MM
%T{Format}Current date/time, using FormatDateTime formatting (see below)%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%s	Current second as SS
%S{Format}Start time in FormatDateTime formatting (used for stop logging)%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%w	IOS8601 week of the year
%tTab character (ASCII 9)%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%T{Format}	Current date/time, using FormatDateTime formatting (see below)
%rCR character (ASCII 13)%nNewline Character (ASCII 10)%gUnique identifier of the associated playback operation%R{Key}Runtime data	%S{Format}	Start time in FormatDateTime formatting (used for stop logging)
%n Newline Character (ASCII 10) %g Unique identifier of the associated playback operation %R{Key} Runtime data	%t	Tab character (ASCII 9)
%g Unique identifier of the associated playback operation %R{Key} Runtime data	%r	CR character (ASCII 13)
%R{Key} Runtime data	%n	Newline Character (ASCII 10)
	%g	Unique identifier of the associated playback operation
%# Total number of current listeners on all encoder connections	%R{Key}	Runtime data
	%#	Total number of current listeners on all encoder connections

Information about the item being played

%а	Artist
%b	Title
%I	Total duration, in sections with fractions
%l{Format}	Total duration, in FormatDateTime format

%L	Total duration, as HH:MM:SS
%d	Actual playback duration, in DirectSound units seconds (for stop logging)
%e	Actual playback duration, in seconds with fractions (for stop logging)
%e{Format}	Actual playback duration, in FormatDateTime format (for stop logging)
%р	Effective playback duration, in seconds
%P	Effective playback duration, as HH:MM:SS
%F	Projected finish time (start time + duration), in FormatDateTimeFormat
%c{Type}	Cue marker, in sections with fractions
%Е	End type
%I	Item type (internal identifier)
%J	Item type (human readable)
%u{Key}	Attribute value
%C	Comment
%U	Internal Database ID
%V	External ID
%k	Icon filename
%К	Icon data as BASE64 string
%1	Filename with path
%2	Filename without path
%3	Filename without path and extension
%4	Filename without path, first character only
%i{Key}	Raw file tag data

Additional variables for playlist/advertising import (filename pattern)

%В	Block number
%2B	Block number, two digits, leading 0
%3B	Block number, three digits, leading 0
%N	Region number
%2N	Region number, two digits, leading 0
%2N	Region number, three digits, leading 0

FormatDateTime

Some variables expect a time format string in Delphi's FormatDateTime notation, see http://www.delphibasics.co.uk/RTL.asp?Name=FormatDateTime

Subsequent items

By using + signs in the variables, you can access the data of the subsequent (next) items in the playlist.

For example %+a retrieves the artist of the next item, not the current one, %++a the artist of the item after the next item, and so on.