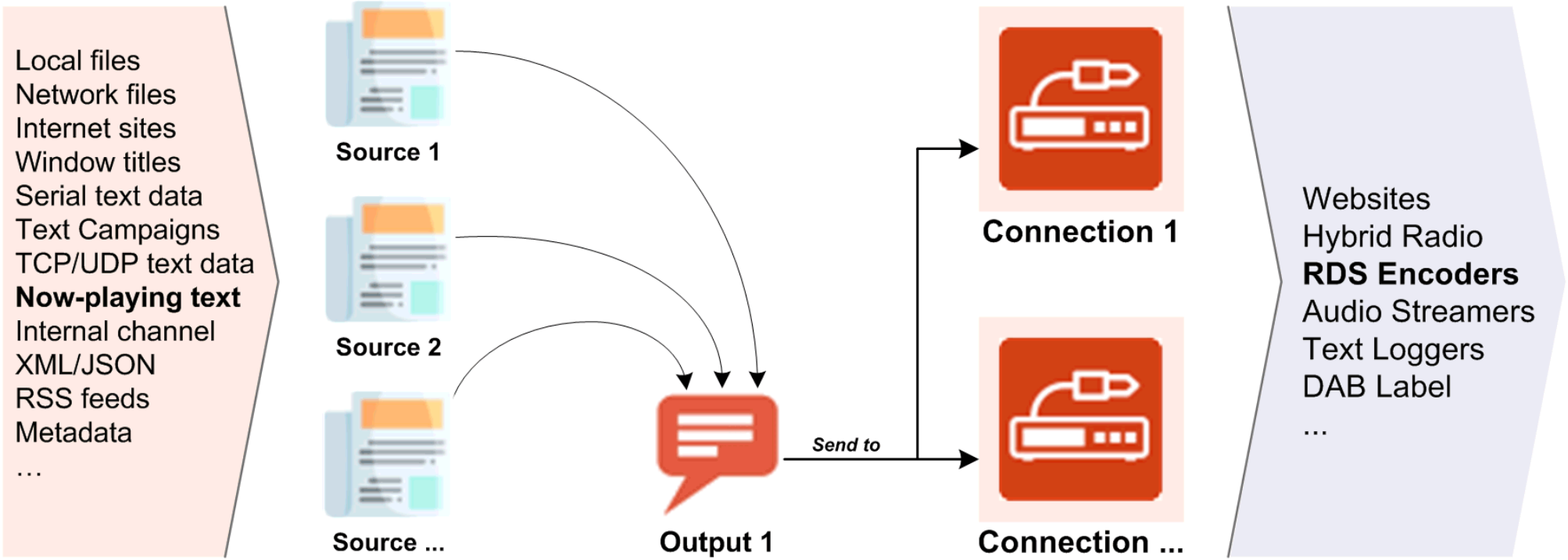
External Text Sources

The External Text tool is the crucial feature of the Magic RDS 4. It reads text or metadata from predefined locations and forwards it to RDS encoders, streaming servers, websites etc. Typical application is providing "now-playing" information about currently playing song and other program-related data to the listeners, optionally alternating with ads, news, weather and other public information. Additional actions can be triggered upon occurrence of keywords in the text, so that other services or devices can be controlled.

Up to 64 text sources may be defined. Several sources may be concatenated in a single output to create a loop of information of different kind. The maximum number of active outputs may vary depending on the license type. Powerful text parsers include text replacing and intuitive support for XML and JSON. The text processing supports exceptions, RegEx and JavaScript. Native script control allows for example specific configuration across different program blocks during the day. Optional RT+ tags (metadata) can be assigned to any text information, so the radio station stays competitive and as modern as other services available.

To open the External Text Sources, go to Tools - External Text Sources in the main menu. Following figure illustrates typical application scheme and usage of the External text tool:



Note: The External Text Sources is a tool which inherently requires permanent connection between the Magic RDS 4 and the device(s) or server(s). This setup is also referenced as 'indirect link' or 'middleware'. It is today's industry standard which offers significantly more sophisticated text processing options and data distribution possibilities, compared to the 'direct link' setup. Please see the device documentation for further comparison between the 'indirect' and 'direct' link. Make sure the destination device or application is configured for accepting and showing the data format selected.

Get it working in a few quick steps

1. In the application main window, add a new ***Connection*** (or skip this step if suitable connection has been already established).
2. In the External text tool, add a new ***Source***. Fill the *File/Class/URL* field according to the location you want to read from. Click on the Test button to verify the function.
3. In the External text tool, add a new ***Output***. Drag the *Source* (created in step 2) and drop it on the *Sequence* field. Tick the *Enabled* box and *Default* box in the *Output Purpose* section. Click on the Save button.
4. The application starts to send the *Source* text to the *Connection*s.

Following sections give detailed description of the configuration options and complete list of all functions available.

TABLE of CONTENTS

Outputs	Radiotext Plus	Dynamic Text Loop
Sources	Inline RT+ Tags	Internal Web Server
Text Replacing	Dynamic Configuration	Javascript
Text Conditions	Dynamic Elements	Examples

Outputs: Add New

The Output is a basic block which holds text sources on the one hand and destination connections on the other hand. The configuration is simple. There's just the list of sources (or a single source) and a list of destinations (connections) where to send the text.

Enabled	This effectively serves as an Output On/Off switch. Disabled Outputs do not perform any operation.
Send on Change	Sends data to the RDS encoder when the source text changes. Enabled by default. Otherwise only the preview is working.
Text Source Sequence	Here you may select which text source(s) will be assigned to the output. Each text source must be defined first. If more sources are selected, these will be used sequentially in a loop. To fill the Sequence, two methods are available: Drag the source icon and drop it on the field. or select the source icon and press the Add Recent button. To change the source order, select the line and press Move Up/Down. To edit the source, double-click on the source icon or select the line and press Edit.
Output Purpose	At least one option must be ticked, otherwise the output will still operate but no text will be sent. The default (first) option will suit to most applications, incl. Radiotext, RT+, text publishing etc. The Dynamic PS option is reserved for use with RDS encoders. Please refer to the device documentation for more details. The Text Conditions option applies to the control based on specified keywords in the text, and will be described later in this chapter. ***Important notes*** *related to RDS encoders:* Before attempting to use the External text tool consider which RDS service you will use for showing the text (Radiotext or Dynamic PS or both). In the application main window, section RDS Content, configure the Radiotext 1 and/or Dynamic PS 1 for each RDS encoder and confirm the settings by Test and Apply buttons. If also Radiotext 2 or Dynamic PS 2 option is present, it should be disabled.

	Using the dynamic (scrolling) PS is restricted in some countries. The manufacturer is not responsible for incompetent use of this feature. The dynamic PS is deprecated since new receivers support Radiotext and Radiotext Plus. Consider suppressing or entirely leaving the dynamic PS usage. The dynamic PS has never been supported by the RDS standard, therefore its behavior on different receivers is unpredictable and it's often confusing for the listener. The PS is a part of your brand and should stay static. The Radiotext is recommended rather. If you're not sure about what option to enable, turn on the Radiotext and keep the Dynamic PS disabled.
Destination Connections	Specifies the text destination(s) by means of the connection name. Each destination occupies extra line. A wildcard * is supported for all 'device' connections. By default, the text is sent to all connections whose kind is not 'Bridge': sendto * The Bridge connections must be specified incl. their name, for example: sendto "RadioWeb" You may simply drag the Connection icon from the application main window and drop it on the list of Destinations. A group of connection is supported as well: sendto group:"Radio One" Optional parameters site, adr, dsn and psn may be specified for encoder's UECP protocol. At least one connection must be added in the application main window, otherwise the text Output will not apply. *Note:* The destinations here have no effect on the Text Conditions, described later in this chapter.

Sources: Add New

The Text Source tells the application where to find the text desired. A separate Source is created for each location (kind) of the text.

[Delete]	Deletes the text source from the database.
[Test]	Shows the text reading result for the source selected. Includes the Text replacing effect.
Source Type *	Several text source types are provided, depending on where the desired content is located:

	Local File The text content is saved in a file which is placed on a local harddisk, virtual disk or local network disk. **Window Title** The text content can be found in window title of some application running on the local machine. For example Winamp shows the "now-playing" information in the window title. If no information can be read (application is currently not running), empty text will be returned. You may use 'N/A text' field for defining a default text or taking an action. **Internet Website** The text content is accessible via HTTP, FTP or secure HTTP internet protocol, from a web server or from an RSS data provider. *Note: If the server cannot provide the file requested, no operation is performed - the source is skipped and another attempt will follow in the next cycle.* **Internal Channel** The text content is created by a different Magic RDS component or by the script (Task Scheduler, Dynamic Text Loop, etc.): Select this option if there's a connection Bridge established with external text source data distribution. An example of use is accepting text from a Virtual port. Select this option to insert a text from the script (Task Scheduler). For example, to update the Source 1 (ID 1), type: *set Source1.InternalChannelText:"new text here"* **TIP:** If you're not sure about location and kind of the text data source, please follow the documentation of your broadcast automation system. All broadcast automation vendors know the Magic RDS and will assist you with correct linking from their software.
[Find File]	Easy way how to find the text source file on local disk or in Microsoft Network local area.
[Find Window]	Easy way how to find the text source window and its title, if the desired application is running.
Coding	Determines the character coding of the source text. Auto - The application will try to detect the coding automatically. This is a default option. ANSI - Force 8-bit coding, equal to the Windows code table. UTF-8 - Force UTF-8 coding, which is a standard coding nowadays.
Process *	Determines the line to be read or the parser to be applied. Has sense only for the *Local File* type and for the *Internet Website* type. For the *Window Title* type it has no effect.

	First line - First line of the file or first RSS item. **Last line** - Last non-empty line of the file or last RSS item. **All lines - one per session** - Each time the source is in the order, a next line is read. **All lines - all per session** - The sequence of all lines is read. Standing on each line takes the time specified by the *Go to next after* parameter. When all lines are processed, the next source takes control or the loop repeats. **Entire file as plain text** - Entire file is read and processed as one block of text **Entire file as XML** - Entire file is read and processed as XML or valid XHTML (typical for XML structure are tags in < >) **Entire file as JSON** - Entire file is read and processed as JSON (typical for JSON structure are { } brackets) **Entire file as script** - In this special case the file is executed as a script (list of tasks), without further processing by the External text tool. For example, this option may be used for sending fixed text messages specific to each RDS encoder in the network. For the tasks syntax, follow the menu item Tools - Execute Script.
Name *	Name of the text source, type any name you like, which well describes the text source.
File/Class/URL *	Depending on the Type choice, this field specifies: path and filename of the local file or Window class name or URL (internet address). For the Local file, wildcards ("*" and "?") are supported (the most recent file found will be read). For example: music_log_??-??-????.txt For the URL, all common formats are supported. Examples of the URL: http://server.com/file.htm https://server.com/file.asp?item=current ftp://login:password@server.com/folder/file.txt
Go to next after	If more that one Source is selected in associated Output, this parameter specifies the time in seconds before going to next Source in the loop. For single Source it has no significant effect. Default value is 60. ***Note:*** During the period, the text is still read repeatedly from the source and checked for any change. The text reload can be suppressed by increasing the 'Keep in cache for' value or by enabling the Output's Transition Optimizer.
Cut everything before/after (and including)	Finds the first occurrence of the text specified and cuts everything before/after, including the searched text.

Cut characters from beginning/end	Cuts fixed number of characters from the text. Performed after previous operation.
Prefix	Prefix placed before the text.
Suffix	Suffix placed behind the text.
N/A text	This text is returned if the source fails, especially: • local file not found • the strings to be cut were not found • the XML/JSON elements do not exist • resulting string is empty: Filling %GOTONEXT% as the N/A text will cause switching to next Source in the Output sequence. ***Note:*** If JavaScript is enabled, the N/A text is not applied on empty strings. Such empty string is passed to the JavaScript instead.
Keep in cache for	The text is placed to internal cache for the time specified in seconds. The text is not read repeatedly from the source until the period elapses. Useful mainly for the Internet Website type or to suppress fast changes of the text. A value of 30 is usually appropriate for text display purposes like news, commercials etc. For the Text Conditions, the value should be 0. **TIP:** If the text being read unexpectedly does not contain latest information, decrease this value.
Expires after	If the text does not change for longer period than specified, corresponding text source will give no output and will be excluded from the loop until the text changes again. This feature may be used for example for sending song details only for a limited time after the song starts playing. It also allows to automatically select right text source if different broadcast software is used during the day - the text which is no longer current will simply expire. Default value is -1 (that means disabled). This parameter has no significant effect if no other Source is placed in the Output sequence. **TIP:** For XML or JSON processing option, the parameter can be updated dynamically if a duration value is included in the file. For example, setting the parameter to %3% takes the value from the element assigned to

	%3%. In this case the duration time format can be h:m:s or m:s or s.
Initial command	If filled, the command is sent to the RDS encoder before the text. For example, it can be used to define different display mode (DPS1MOD=x) or number of transmissions (DPS1REP=x,CLR) for each source. Only one command can be inserted. For experts only, keep empty otherwise.

Following options apply only if the XML processing is selected:

Element name or full path	Element name or full XML path to the values of interest. Up to four elements are allowed. For example, let's suppose an XML file: <Current> <Title>Can You Feel It</Title> <Artist>Larry Heard</Artist> </Current> Corresponding paths would be: /Current/Title /Current/Artist ***Hint:*** The XML language is case sensitive. **TIP:** Click on the Find Element button for intuitive single-click selection of desired element.
Output template	Defines the output text and determines where the XML element values appear in it. For example: Now Playing: %1% by %2% where %1% is finally replaced by the song title and %2% is finally replaced by the artist name.

Following options apply only if the JSON processing is selected:

Element name or full path	Element name or full JSON path to the values of interest. Up to four elements are allowed.

	For example, let's suppose a JSON file:
	
	{
	"Current": {
	"Title": "Can You Feel It",
	"Artist": "Larry Heard"
	}
	}
	
	Corresponding paths would be:
	
	/Current/Title
	/Current/Artist
	
	Hint: The JSON language is case sensitive.
	
	TIP: Click on the Find Element button for intuitive single-click selection of desired element.

Output template	Defines the output text and determines where the JSON element values appear in it. For example: Now Playing: %1% by %2% where %1% is finally replaced by the song title and %2% is finally replaced by the artist name.

* marked items are obligatory

Note: Leave unused fields empty.

RT+ (Radiotext Plus)

This chapter applies to the RDS Radiotext service and the DAB+ Dynamic Label service as well.

The RT+ is an enhancement of the Radiotext (RT). It is designed to let the listener take additional benefit from the Radiotext service by enabling receivers to offer direct access to specific elements of the Radiotext. These elements anyway carried in the Radiotext, are identified by their class code, length and location within the Radiotext. The receiver must be equipped with the RT+ function (also called

"tagging") to take advantage of this feature. The RT+ module implemented in the Magic RDS supports individual tag assignment to each text source, and furthermore to each kind of content, based on sets of templates.

Please note that Radiotext Plus is an important service that improves the listener's experience. It is essential that the listener feels that radio is still a modern medium. Turn on and use Radiotext Plus whenever possible. If your RDS encoder does not support RT+ or ODA, replace it with another one, preferably with XCMD protocol support.

Important: The RT+ service is only a marker for existing text segment(s) in the Radiotext. Thus, the RT+ can only carry a text which is already contained within the Radiotext.

Enable RT+	Enables the RT+ feature for selected text source. The RT+ data will be sent together with the Radiotext.
Assign RT+ Class to Element (XML/JSON only)	Assigns the RT+ tag class directly to a particular XML/JSON element, as read from the source.
Define RT+ Template(s)	Locates the RT+ tags inside the text (if each tag lies between static text sections) or marks entire text as a tag. You can define one or more templates (sorted by priority) for the RT+. If the text matches the template, corresponding dynamic text segments are marked as the RT+ tags. The comparison is case-sensitive. **Example 1** Final text: Now playing: Find a Way by Offaiah Suitable template: Now playing: <item.title/> by <item.artist/> **Example 2** Final text: New Year fireworks cancelled to protect walrus Suitable template: <info.news/>

	Examples of advanced use The tag element accepts optional regexp (regular expression) parameter which effectively restricts the set of matching text contents: <info.news regexp="BBC News.*" /> <phone.studio regexp="707 123 456" /> etc.
RT+ Preview	You can verify your RT+ settings here. The preview is updated on each Radiotext update or clicking on the Test button. ***Note:*** Maximum Radiotext length is 64 characters. For longer text, the RT+ tags may be truncated.

Note: The RT+ can service can handle up to 2 different tags (for example artist and title). If more tags are assigned or defined, only first two tags will apply.

Before attempting to use the RT+ feature make sure the Radiotext is properly read from the text source, showed on the receiver's display and contains the text information expected.

RSS support

Enable RSS Support	If enabled, the source text is first tested for RSS **2.0** structure presence and if this is found, then it is parsed as RSS 2.0.
Target	Read title or description lines?
Limit	Limits the number of lines (items) read from the RSS channel.

Important: One text source may be assigned to more services (like Dynamic PS 1, Radiotext 1, Text conditions etc.). But if there are more text sources in use, it's preferred to establish a separate set of text sources for each service (even though there will be more text sources reading the same file or location). This way allows to control additional differences between the services and ensures that values like 'Keep in cache for' will work exactly as expected.

Text replacing

A strong text replacement tool with support of regular expressions and some script commands. Compared to the JavaScript, it does not require any programming skills and brings some further features. May be used for replacing of single characters, words or sentences as well as for text filtering, advanced text processing and conditional selection of text source. See the Examples for more details.

Each row of the table represents one item. In the column labeled "Change", enter the plain text, character, or regular expression that you want to replace or that represents the corresponding condition. In the column labeled "to", enter the desired text after replacement, or one of the available commands, or a combination of both. The column labeled "Filter" is optional. It is used to select the text source, output, or XML/JSON element to which the item should apply. If nothing is filled in, the item applies to all sources, outputs, and elements.

Select a cell and click on right mouse button to insert or delete a row, to insert or delete a command, and to fill the Filter.

Basic text processing sequence:
Reading from the source → Text cutting → (JavaScript) → **Text Replacing** → (Adding prefix/suffix) → resulting text

For XML/JSON sources, the text replacing is applied to individual elements (1 to 4), not to the template or the entire composed text.

Text conditions

The Text conditions is a general purpose tool whose job is to perform predefined action, based on occurence of keywords or expressions in the text. The trigger moment occurs when the condition flips.
Before using the Text conditions, tick the "Text Conditions" option in appropriate Output or create a separate Output for this purpose. Depending on the license type, you may create more independent Outputs and related conditions. If some line needs to be assigned only to a specific Output, write that Output's name into the 4th column.

A few application examples of the Text conditions:

- Toggle RDS TA (Traffic Announcement) if specified keywords are found in the "now-playing" text provided by the broadcast automation system
- Launch any application or Windows batch file if specified expression is found on some website
- Send a text command to some Connection or website if some application window title is detected on the local PC
- Perform some action if specified string disappears in some text

- Run/Pause/Stop the audio mp3 recorder if specified keywords are found in the text
- Add or remove lines in the Dynamic Text Loop
- etc.

See also the Examples below.

The command to be executed is expressed by means of the Magic RDS 4 scripting repertoire. More commands are accepted, if they are separated by ;
For complete list of commands, follow the scripts.pdf help file.

Tip: In some cases it is preferable to use separate text Output for this feature rather than share it with Radiotext or Dynamic PS. This allows to set individual properties although the source file or location may be the same. Especially the 'Keep in cache for' value should be set to 0 when used with the Text conditions to allow immediate response when the source text changes.

Note: The Text conditions tool is **not** directly compatible with older Magic RDS 3. The command syntax is different. See the Examples below.
Note: The Text replacing feature is not used for the Text conditions (the comparison is made just before applying the Text replacing). The Text conditions tool however still supports regular expressions (using the %REGEXP% prefix) and the text can be preprocessed by the JavaScript.
Note: In the Basic license, the Text Conditions work only in demo mode and can be enabled only for a limited time.

JavaScript

Once the text is read from the Source, it can be modified by the JavaScript (JS). This option is suitable for advanced users and specialists.

Note: This option is not available in Windows XP/Vista.

Important:
The JS is only used for the text modification purposes, using the string handling functions. Other kind of use may not be supported by the environment.
The JS is applicable only in the text box which is explicitly marked as 'JavaScript'. In other cases, the 'script' is expected to be the Magic RDS 4 proprietary script (see the scripts.pdf for details).

The JS execution time is limited to 2 seconds to protect the application from user JS code errors like infinite loop or infinite recursive calls.

Type the JS commands directly into the JavaScript text box. The JS shoud be Disabled during the script editing. The JS must end by *return* command, followed by the output text or variable.

The simplest JS has only single line:
return '%TEXT%';
which effectively returns the input text with no modification made.

Once you assign the %TEXT% dynamic tag to a variable, you may modify the variable content as you wish:
var t = '%TEXT%';
...
return t;

The Insert button offers further dynamic tags and control elements.

To apply the JS code only to a specific Source, use this template:
var srcName = '%SRC_NAME%';
if ((srcName == 'Source 1') || (srcName == 'TEST')) { ... }

To apply the JS code only to a specific Output, use this template:
var outName = '%OUT_NAME%';
if ((outName == 'Output 1') || (outName == 'TEST')) { ... }

The text processing sequence is as follows:
Reading from the source → Text cutting → **JavaScript** → Text Replacing → (Adding prefix/suffix) → resulting text

Dynamic Text Loop

Advanced users may achieve various text automation effects using the Dynamic Text Loop. The Dynamic Text Loop is based on a set of lines - text messages (for example station slogans, ads and program announcements) which are sent out in a loop. The Dynamic Text Loop adds an API for individual control of each line. The Magic RDS can automatically add lines, remove selected lines and suspend some lines for a predefined time.

The Dynamic Text Loop is characterized as follows:

- Each Text Source can handle one Dynamic Text Loop
- The text loop mode is available if the Source type is set to 'Internal channel'
- The text loop mode is set up automatically for a particular Source upon applying the *textloop* command
- The Dynamic Text Loop can be combined with other Sources by putting more Sources to the Output
- The lines are **not** saved on the application exit. To create a set of default lines, use the menu option Tools - Presets and type the script for that, or use the Task Scheduler.

Quick guide for setting up the Dynamic Text Loop

1. Add a new text Source, type appropriate name for this source, for example Source 1
2. Put this Source to the Output
3. In Tools - Execute script, establish the text loop using the *textloop* command with *action* parameter set to *add:*
   textloop source:"Source 1" action:"add" name:"Line1" text:"My first text message"
   textloop source:"Source 1" action:"add" name:"Line2" text:"My second text message"
4. Use the Task Scheduler and Text Conditions tools for further working with the lines

Notes:
For complete API reference, please follow the Help document scripts.pdf.
To see current Dynamic Text Loop content, select View - Debug - Spreadsheets in the main menu.

Inline RT+ tags

Any text message from any plain text source can contain inline RT+ tags. Such tags are prioritized, i.e. if the text message contains inline RT+ tags, any other RT+ tag assignment is ignored.

Example of the text message with inline RT+ tag:
Call us: <Phone.Studio>123-456-789</Phone.Studio>

The tag processing is case insensitive. The tags are converted into RT+ markers and then are removed from the text. The tags are subject to the Text replacing like any other text. You may place two tags at maximum to the text.

List of supported tags:
Item.Title, Item.Album, Item.Tracknumber, Item.Artist, Item.Composition, Item.Movement, Item.Conductor, Item.Composer, Item.Band, Item.Comment, Item.Genre, Info.News, Info.News.Local, Info.Stockmarket, Info.Sport, Info.Lottery, Info.Horoscope, Info.Daily_Diversion, Info.Health, Info.Event, Info.Scene, Info.Cinema, Info.TV, Info.Date_Time, Info.Weather, Info.Traffic, Info.Alarm, Info.Advertisement, Info.URL, Info.Other, Stationname.Short, Stationname.Long, Programme.Now, Programme.Next, Programme.Part, Programme.Host, Programme.Editorial_Staff, Programme.Frequency, Programme.Homepage, Programme.Subchannel, Phone.Hotline, Phone.Studio, Phone.Other, SMS.Studio, SMS.Other, Email.Hotline, Email.Studio, Email.Other, MMS.Other, Chat, Chat.Centre, Vote.Question, Vote.Centre, Reserved_54, Reserved_55, Private_56, Private_57, Private_58, Place, Appointment, Identifier, Purchase, Get_Data.

Important! RDS standards prohibit nesting of different tags. The tags cannot overlap, neither fully or partially. For example, this is **incorrect** and not supported by the Magic RDS: <Stationname.Long><Stationname.Short>KIWI</Stationname.Short> FM 106.0</Stationname.Long>

Dynamic configuration of the External Text Sources

Want to change and control the text sources behaviour automatically during the day, upon a specific event? The External Text Tool supports configuration via simple script commands. Such commands can be executed from the Task Scheduler, Text Conditions or by other methods, as described in the document scripts.pdf.

Example 1: Temporarily disable the external text source, show a static Radiotext instead

set Output1.Enabled:"0"
send connection:"" command:"TEXT=Type the text message here"*

To return back to normal operation:

set Output1.Enabled:"1"

Hint: A unique ID number is assigned to each Output in the External Text Sources. Such ID can be found in the header, upon clicking on particular Output icon.
To enable the Output with ID=1, type *set Output1.Enabled:"1"*, to enable the Output with ID=2, type *set Output2.Enabled:"1"*, etc.

Example 2: Switch from Output 1 to Output 2

set Output1.Enabled:"0"
set Output2.Enabled:"1"

Example 3: Switch the source in the Output 1 sequence

set Output1.SrcSequenceItem1Checked:"0"
set Output1.SrcSequenceItem2Checked:"1"

Hint: Each Output contains a sequence of text sources. That sequence may contain a single item or several items. Only the "checked" items (sources) are applied.
For the purpose of the *set* command, each item is identified by its sequence number: *SrcSequenceItem1* for the first item, *SrcSequenceItem2* for second item, etc.

Example 4: Changing input file for reading the now-playing information

set Source1.Path:"D:\radio2\nowplaying.txt"

Hint: If you switch between different text sources throughout the day and they only differ in the file they read from, then consider using the example above, as you only need a single text source.

Dynamic elements in the text

When a dynamic element is placed anywhere in the text, actual value is sent to the encoder when the Radiotext or Dynamic PS is served out. Syntactically, the dynamic elements are enclosed between two % characters. The dynamic elements can occur anywhere, in the prefix, suffix, file content etc. **The most important thing about the dynamic elements is that this function belongs to the External Text Tool.** The replacement occurs just **before** the text is sent to the encoder. Some encoders may support further dynamic elements which are updated by the encoder itself.

Example: The dynamic element **%DATETIME:dd-mm-yy%** is replaced by current date before sending the final text to the RDS encoder. The date format can be specified.

List of dynamic elements

DATETIME:	
d	Displays the day as a number without a leading zero (1-31).
dd	Displays the day as a number with a leading zero (01-31).
ddd	Displays the day as an abbreviation (Sun-Sat).
dddd	Displays the day as a full name (Sunday-Saturday).
m	Displays the month as a number without a leading zero (1-12). If the m specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.
mm	Displays the month as a number with a leading zero (01-12). If the mm specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.
mmm	Displays the month as an abbreviation (Jan-Dec).
mmmm	Displays the month as a full name (January-December).
yy	Displays the year as a two-digit number (00-99).
yyyy	Displays the year as a four-digit number (0000-9999).
h	Displays the hour without a leading zero (0-23).
hh	Displays the hour with a leading zero (00-23).
n	Displays the minute without a leading zero (0-59).
nn	Displays the minute with a leading zero (00-59).
am/pm	Uses the 12-hour clock for the preceding h or hh specifier, and displays 'am' for any hour before noon, and 'pm' for any hour after noon. The am/pm specifier can use lower, upper, or mixed case, and the result is displayed accordingly.
a/p	Uses the 12-hour clock for the preceding h or hh specifier, and displays 'a' for any hour before noon, and 'p' for any hour after noon. The a/p specifier can use lower, upper, or mixed case, and the result is displayed accordingly.
'xx' or "xx"	Characters enclosed in single or double quotation marks are displayed as such, and do not affect formatting.

Further dynamic elements can be supported on request.

Text publishing via internal web server

Current text generated by the External text tool can be directly accessible via the internal web server. This feature requires Activation or Standard or Full license. For example, you can include the text to your web site using IFRAME element.

1. Make sure the web server is enabled (Option - Preferences - Web Server).
2. Create a custom web page containing this dynamic tag: **%output<ID>_ext**, where <ID> is the Output ID in range 1 to 32. See the example below.
3. Place that file to a folder which belongs to the internal web server file system. This would be the WWWRoot/usr/ directory in the application folder or the usr directory as specified in the Preferences. Access to this folder does not require login.

```
<html>
<head></head>
<body>
<p>Now playing: %output1_ext</p>
</body>
</html>
```

Examples

Text Replacing

For this example, consider a text file generated by your broadcast automation system. Fill the following to the Text replacing grid and see the effect:

Change	to	Filter (optional)	Effect:
&	and		Character "&" can't be displayed on old receivers, so we replace it by a word that can be always displayed.
ö	%KEEPCASE%oe		Character "ö" can't be displayed on old receivers, so we replace it by letters that can be always displayed.

%S%jingle	%CLEAR%%END%		Don't show anything during playing a jingle, stop processing the text.
%SC%Background 1	%CLEAR%Now speaking John Smith%END%		Assign a custom text to a background sound, then stop processing the text.
produced by	%TERMINATE%		Remove anything what is written after the 'produced by' words (including).
%SC%Background	%CLEAR%%SOURCE2TEXT%%END%		Assign a text from another text source to the background sound, then stop processing the text.
%REGEXP%^.{0,2}$	%CLEAR%Default message		Replace all strings shorter that 3 characters by a default message.
%REGEXP%#...			Remove all substrings like #GHI, #RPT or #CCH
%REGEXP%.+	%UPPERCASE%	SOURCE1	Any text from Source 1 will be converted to UPPERCASE letters. Does not affect other sources.
%REGEXP%.+-	%TOUPPERCASE%		Converts the text part before - to UPPERCASE letters in single-line text information.
%REGEXP%-.+	%TOTITLECASE%		Converts the text part behind the - to Title Case in single-line text information.
%REGEXP%^.{0,30}	%CLEAR%%MATCHEDSTR%	SOURCE2	Limits the Source 2 text length to 30 characters.
%REGEXP%(.*) - (.*)	%$2% - %$1%		Swaps the artist and title in single-line text information.
%REGEXP%^-|^ -|-$|- $	%GOTONEXT%		Skips the text source if artist or title is missing in single-line text information. (format: 'artist - title')
%REGEXP%^$	%GOTONEXT%		Skips the text source if the text is empty (does not apply to the 'N/A text')
%REGEXP%^.	%TOUPPERCASE%		Ensure that the first letter of a text string is always uppercase
%REGEXP%^$	%CLEAR%%4%	ELEMENT1, ELEMENT2	If XML/JSON element %1% or %2% is empty, don't use the Output Template and take the text from element %4%.
%REGEXP%^(?!.*play).*$	%GOTONEXT%	SOURCE9, ELEMENT3	If Source 9 is XML/JSON type and element %3% value is **not** 'play', skip the Source entirely.

Note: When trying the examples, please take care of the space characters.

Prefixes accepted in the *CHANGE* (expression) column:

%C% prefix means that the replacing will be case-sensitive. Default is case-insensitive.
%S% prefix means that the text must start with expression filled.
%SC% prefix means that the text must start with expression filled. The matching will be case-sensitive.
%REGEXP% prefix means that the string will be processed as a *regular expression*.
The prefixes above cannot be combined (used together)!

Commands and dynamic elements accepted in the *TO* column:

%CLEAR% command means that whole text (not only the expression found) will be deleted and replaced by the text written behind the command. The text source's prefix/suffix is not added in this case.
%TERMINATE% command terminates the text on a first occurrence position of the expression. Any text behind the expression found (and including it) will be deleted.
%x% (for XML or JSON source only): if the expression is found, it is replaced by a text from corresponding XML/JSON element. The x is in range 1 to 4.
%SOURCExTEXT% element means that if the expression is found, it is replaced by file content from another text source, the x is the Source ID in range 1 to 64.

%UPPERCASE% command means that if the expression is found, entire text is converted to UPPERCASE.
%TITLECASE% command means that if the expression is found, entire text is converted to TitleCase.

%KEEPCASE% command can be used for single character or word replacement, the resulting text segment is automatically converted to lowercase, uppercase or titlecase, according to the original text being replaced.

%TOUPPERCASE% command means that if the expression is found, that part of the text is converted to UPPERCASE. Applies to RegExp only.
%TOTITLECASE% command means that if the expression is found, that part of the text is converted to TitleCase. Applies to RegExp only.
%MATCHEDSTR% element represents the matched string (without conversion). Applies to RegExp only.
%$x% element represents the matched RegExp group (substitute). The x is in range from 1 to number of groups. Applies to RegExp only.

%END% command means that next rows are ignored if the expression is found.
%GOTONEXT% command causes the text source to be skipped in the Output sequence. Next source in the loop will apply (if exists).

Keywords accepted in the *FILTER* column:

SOURCEx keyword means that the line will be processed only for specified text Source(s), the x is the Source ID in range 1 to 64.
OUTPUTx keyword means that the line will be processed only for specified text Output(s), the x is the Output ID in range 1 to 32.
ELEMENTx keyword means that the line will be processed only for specified XML/JSON elements, the x is the element number in range 1 to 4.
Separate more keywords by ,
If you want to temporarily disable a row, fill in a non-existent Source, for example SOURCE0.

Control words are case-sensitive and must always be entered in uppercase.

Text Conditions

Check corresponding text source file/window, fill the following to the Text Conditions grid and enable the feature:

If found	then execute	else execute	Output name (optional)
jingle - traffic start	send connection:"*" command:"TA=1"		
jingle - traffic end	send connection:"*" command:"TA=0"		
background - news	send connection:"*" command:"PTY=1"	send connection:"*" command:"PTY=3"	
%REGEXP%Temperature: (28|29|3|4|5)	open "C:\Program Files\MyApp\ac_on.bat"		Weather
STUDIO2	send connection:"*" command:"PS=STUDIO2" set Output1.Enabled:"0" set Output2.Enabled:"1"	send connection:"*" command:"PS=STUDIO1" set Output1.Enabled:"1" set Output2.Enabled:"0"	AutomationCtrl

Typically, the expression in the 'If found' column is a part of the audio file name/tag as stored in your broadcast automation system. By default, the matching is not case-sensitive.

For complete list of commands to be executed, follow the Execute script tool, the Task Scheduler tool, the Terminal and the RDS encoder's Technical manual.