Data Repeater

Installation & Operations Manual

Information	Validation Codes	Website
info@bsiusa.com	codes@bsiusa.com	www.bsiusa.com

Sales
Phone: 541-338-8588
Fax: 541-338-8656
sales@bsiusa.com

Technical Support
541-342-5250
support.bsiusa.com

Broadcast Software International
909 International Way
Springfield, Oregon 97477 USA

31 January 2013 – Data Repeater Manual 1.0.4

Table of Contents

In addition to the table of contents, we have included several other navigational aids in this manual. Special icons will alert you to items of particular significance:

A useful tip or cool feature.

Take note, be cautious, and read carefully.

ALERT - Extreme caution.

Glossary

IP Address

Each computer on an IP network is uniquely identifiable by any device on the network by an address, called the IP Address. There are two IP address structures in use today: IPv4 and IPv6. IPv4 is still the most widely used by far (Data Repeater 1.0 supports only IPv4). IPv4 is a 32bit address, while IPv6 is a 128bit address, so IPv6 supports several orders of magnitude more devices on a network.

TCP

A communications protocol (language) used to transfer data over an IP network. TCP stands for "Transmission Control Protocol", and is a protocol developed for the internet to get data from one network device to another. TCP uses a client/server handshake process to ensure that the data sent is the exact data received.

UDP

Another communications protocol (language) used to transfer data over an IP network. UDP stands for "User Datagram Protocol". UDP is similar to TCP in that it transmits data from a server to a client, but there is no handshake process and no process that verifies the data sent is the data received.

Port (IP)

With IP networks, the port is a connection point. Where the IP address identifies your computer on your network, ports are used to identify specific connections within your computer. With TCP, a TCP Server will listen on a specific port number, and a client will try to make a connection on that same port. UDP functions in a similar manner, but there isn't a server and client, per se. Many TCP and UDP connections from multiple programs running on your computer can be using the network connection at the same time, so splitting up the connections by port allows them all to run at the same time without confusing each other.

HTTP Call

An HTTP Call is normally used with web browsers to request a web page. Many Stream Encoders have used the HTTP Call function as a means to send artist & title metadata from your automation system to the stream encoder so the data can be forwarded to your listeners to display the currently playing song's artist and title information.

HTML

HTML stands for Hyper Text Markup Language. It is a set of rules and codes that are included within a text file to create graphical elements on web pages. In an HTML document, in addition to the text you see on screen on a particular website, there are codes that dictate text orientation, justification, location, background and foreground color, borders, images, and more.

XML

XML stands for Extensible Markup Language (so why isn't it EML?). It is a set of rules for encoding text documents in a format that is easily read by a computer. It employs hierarchy using "nodes", and encapsulates data within starting and ending tags that identify what that data is.

Command Line

The Command Line is a method of inputting instructions to your operating system (DOS or Windows) using text. Unlike double-clicking on an icon to launch an application, starting an application (aka

executable) using the Command Line allows you to also include other commands with your application's startup using what are called "command line switches". These switches are extra instructions to launch your application with initialize settings, instructions to load in a specified state, or to give the application extra information for performing what ever task that application is supposed to perform.

Serial Communications

The Serial port is a hardware port built in to your computer (or from USB to Serial Port adapters, too). It is a port type widely used for hardware devices, such as RDS encoders, information displays, audio switchers, or simple bi-computer communications. Serial devices are easy for hardware manufacturers to implement, and easy for end users to configure. Text and binary data are easily transferred between a sending device and a receiving device in small bursts, but the communication speed is slow for large amounts of data and the two devices must be in relatively close proximity to each other.

Data Parsing

For a computer to be able to find specific information in a document or set of data, it must have instructions on how to deconstruct the data to find the specific information it is looking for. The process of deconstructing the data with given instructions is called parsing. The instructions for properly parsing a particular set of data need to include precise information on how to identify the start and end of each portion of data it is looking for, and what that piece of data is.

Category

A type of specification assigned to all audio file formats used in the broadcast industry. Used as a means of grouping a collection of similar files.

Dongle (AKA Hardware Key)

A dongle, or hardware key, is a small USB hardware device that attaches to a computer and holds uniquely identifiable validation information. Data Repeater can be validated with or without a hardware key.

Template

A template is a means of creating a generic format for data to be inserted into. Templates are created with an ASCII Text editor, such as notepad.exe, and consist of text that instructs Data Repeater how to format your data before sending it to a destination. The template is created to look exactly like the data format you desire, except that rather than containing actual artist, title, album, or other song-specific information, it contains meta variables. The meta variables in the template are replaced with the actual artist, title, etc. information that is received by Data Repeater, and the non-meta variable text from the template remains as-is. The "merged" version of the template, containing the specific song data received, is then sent to the destination.

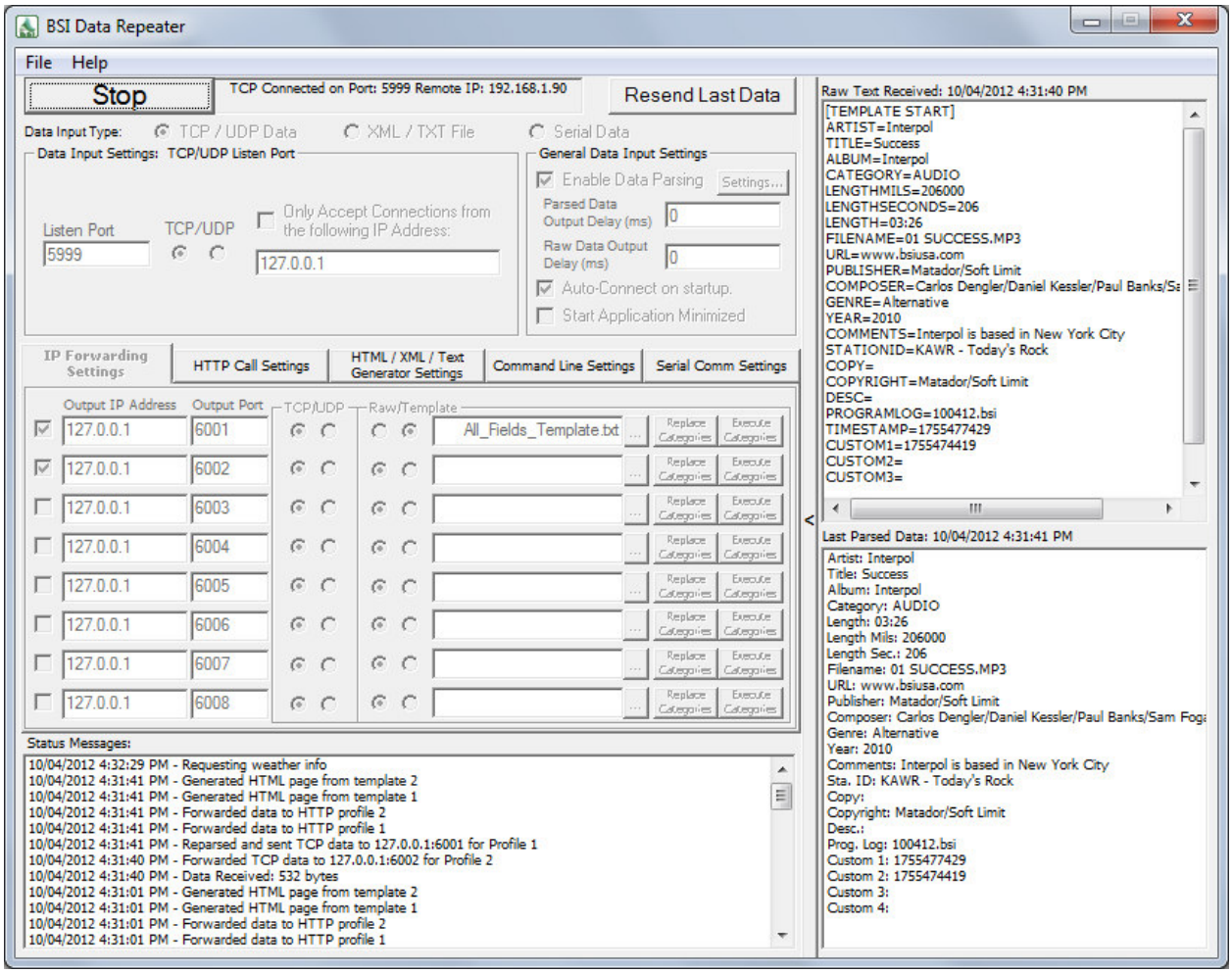
Meta Variable

A meta variable is a specific string of characters that Data Repeater can recognize in a Template that signifies a place-holder for a piece of information. For instance, in a TCP/UDP template, the meta variable %ARTIST% will be replaced by the actual artist name received before the "merged" template is sent on to the data destination.

Introduction

Overview

BSI Data Repeater allows song meta-data received via TCP, UDP, serial, or updated to a text file to be output to a multitude of destinations. Data can be forwarded to 8 TCP/UDP, 7 HTTP Call, 8 Serial destinations. The Data Repeater can also generate 3 HTML pages using user created HTML page templates. Up to 6 different command lines can also be executed, with received data included in the command line using meta-variables, too!



PC Requirements

Pentium 4 1.8GHz PC or better (Core 2 Duo or Core i processor recommended)

Microsoft Windows XP or Windows 7 32bit operating system

1GB (Windows XP) or 2GB (Windows 7 32bit) recommended

15 Mb of hard drive space to store program

Features

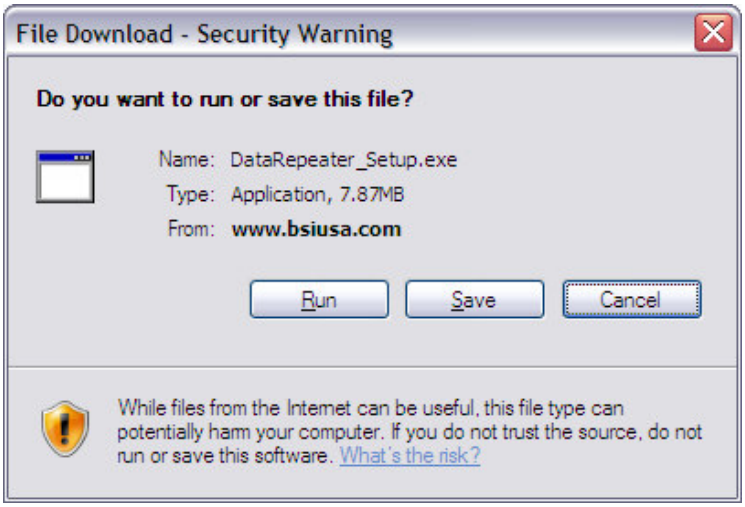
- Receives audio file data via TCP, UDP, Text File (including XML) on hard drive, or serial data
- Easily route your automation system's audio file data to multiple destinations, each of which can have its own data format using user defined output templates
- Forward raw received incoming data to up to 8 TCP/UDP destinations
- Simultaneously output reformatted data to the following destinations:
    - Up to 8 TCP/UDP destinations (total TCP/UDP destinations shared with raw forwards)
    - Up to 7 HTTP Call destinations including meta data within URL called
    - Up to 3 HTML/XML/Text file documents including meta data within document
    - Up to 6 Command Line applications including meta data within command line
    - Up to 8 Serial Strings, on serial ports 1-8 (if available) including meta data within string
- Template based reformatting so you can define different output formats for each data output type
- Auto-update your web page(s) using the HTML Generator to create audio-file specific web page to display the currently playing artist/title info on your website
- Automatically downloads the current weather info for your location to include in HTML Generator web pages to let your listeners get the current weather conditions from your web page

Software Installation

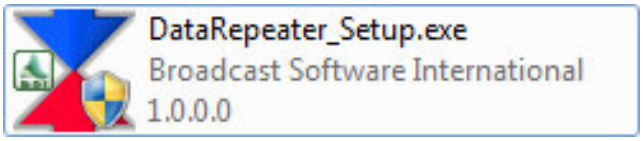
BSI Data Repeater can be downloaded via the Internet from www.bsiusa.com, or installed from the BSI Install CD.

Installing from the Internet

On the BSI Web Site follow the links to the software page and click the download link. When the following (or similar, depending on your web browser) Security Warning appears, click **Save** and select a folder on your hard-drive in which to store the **DataRepeater_ Setup.exe** installer file.



When the download is complete, use Windows Explorer to navigate to the folder in which you saved the installer file. If your Windows folder view is set up to show "tiles", it will look something like this:



Double-click the file to launch the setup process and follow the on-screen prompts as outlined on the next few pages.

Installing from the BSI Install CD

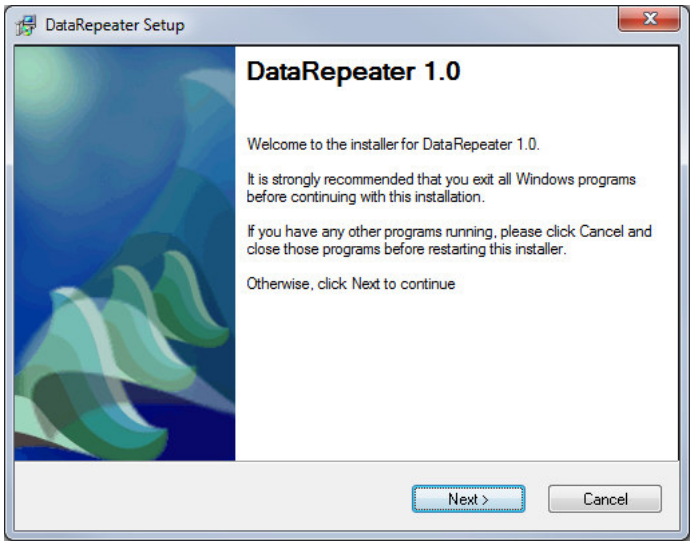
The BSI software installation CD is designed to automatically run when inserted into a CD or DVD drive. If you have disabled this Windows feature, use Windows Explorer to navigate to your CD or DVD drive and double-click the Install.exe icon to launch the CD setup.

Use the drop-down box to select Data Repeater and then click the "Install" button (shown in red at right).

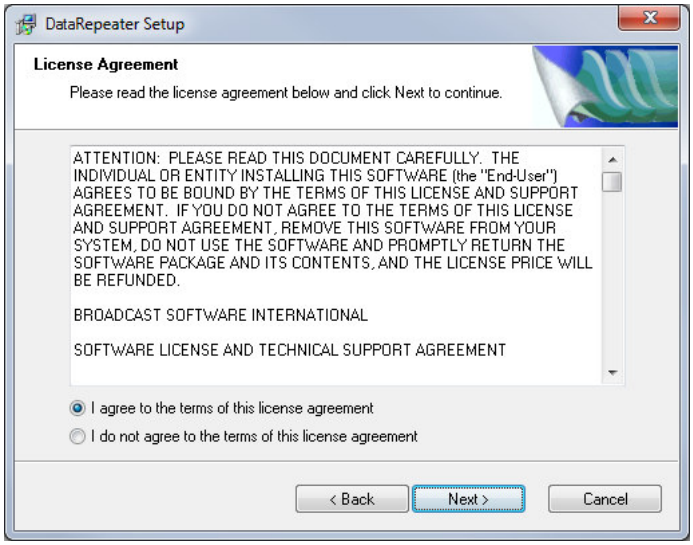
Follow the on-screen prompts as described below.

Installation - all versions

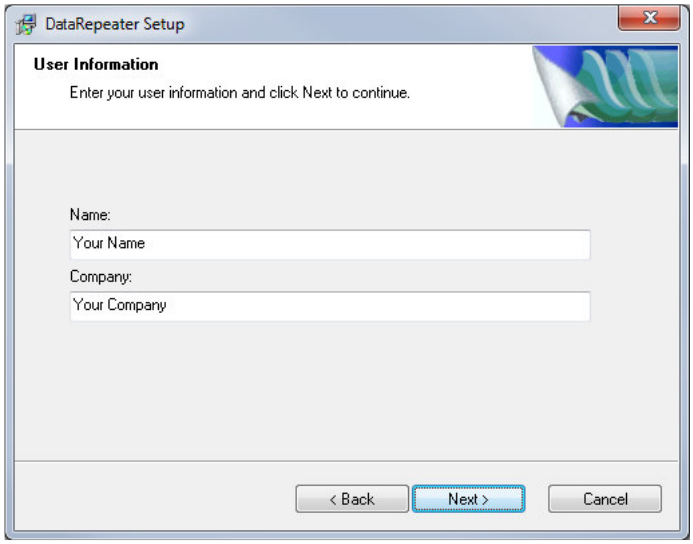
The initial installation introduction page will be displayed. Click **Next >** to continue.



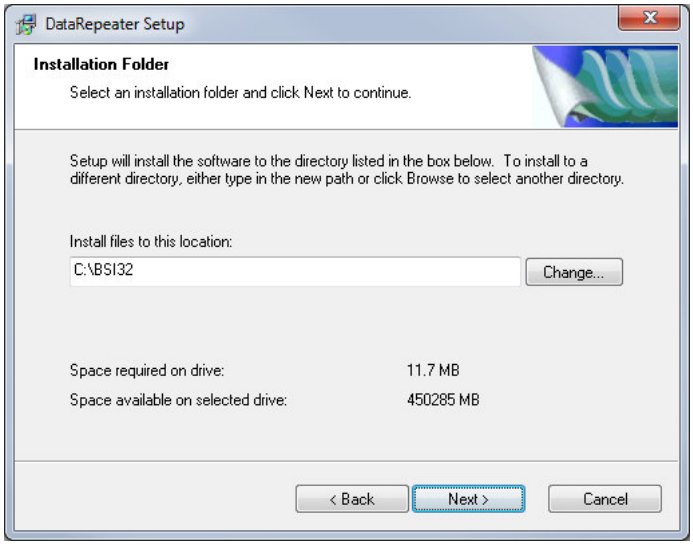
Read through the license agreement and when satisfied, choose the 'I agree to the terms of this license agreement' option, then click **Next>**.



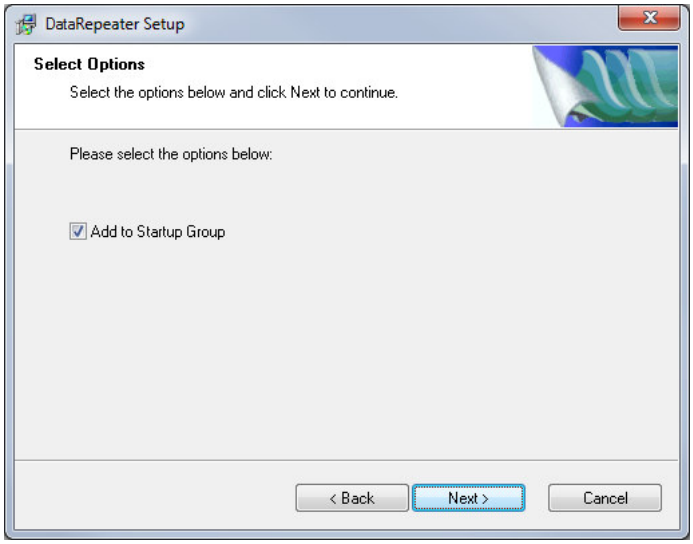
Enter your Name and Company as directed and click **Next >**.



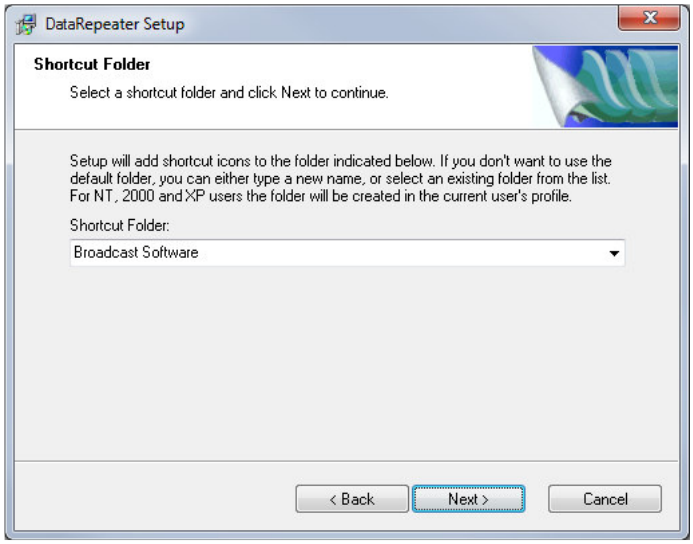
For most installs, leave the install location set as shown. If you wish to change your install location, click on the **Change…** button and browse to your desired install location. Click **Next >**.



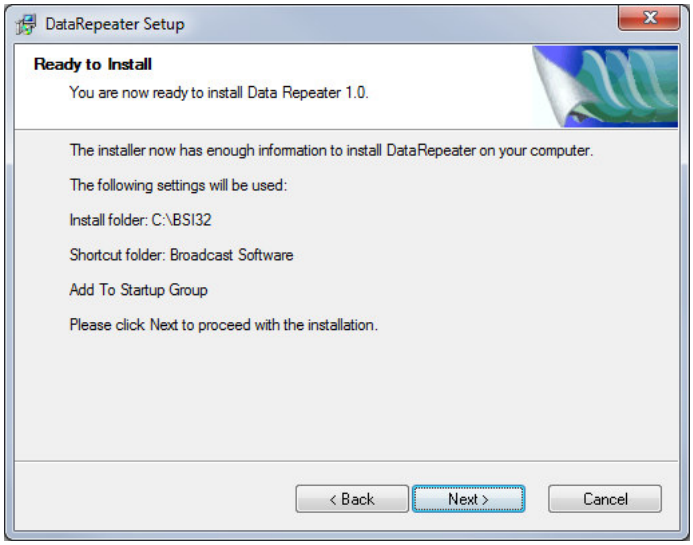
If you would like the Data Repeater to automatically start when Windows starts up, place a checkmark into the **Add to Startup Group** option. Click **Next >**.



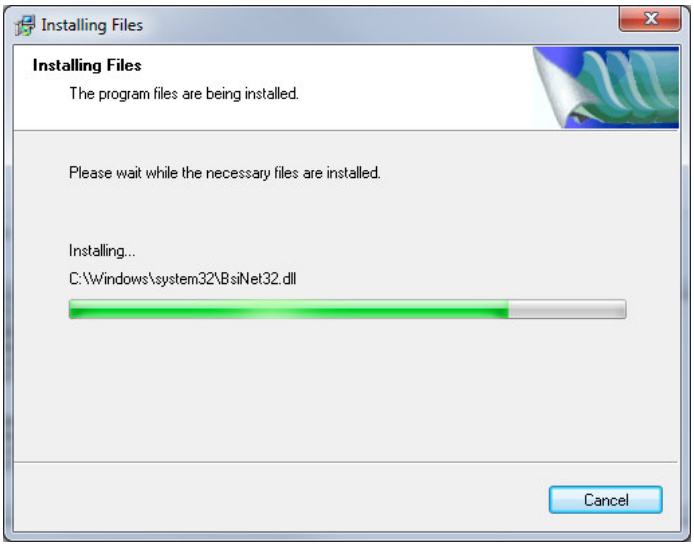
The installer will place a shortcut into the shortcut folder shown. For most installations, this does not need to be changed. If you have already created an alternate shortcut folder in the Start menu in which you would like to place the Data Repeater's shortcut, select it from the drop down list here. Click **Next >**.



The installer now has all the information it needs to complete the installation process. Click **Next >**.

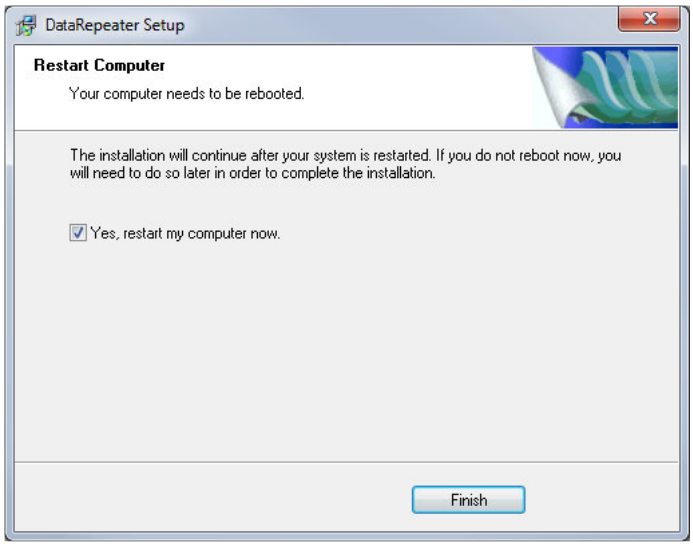


Please wait while the necessary files are installed on your computer.



When Setup is complete, you will need to restart your computer. Click **Finish** to restart now.

If you don't want to restart your computer yet, uncheck the **Yes, restart my computer now** option and click Finish to restart later.



Now that the BSI Data Repeater is installed on your computer you can launch it from the desktop icon or the Data Repeater shortcut in the Broadcast Software shortcut folder in the Windows Start menu (**Start | All Programs | Broadcast Software | Data Repeater** unless you placed it into a different shortcut folder during installation).

When first installed, Data Repeater will run in demo mode and time out after a short demonstration period because it is not yet registered.

When you're ready to register and validate your copy of Data Repeater, follow the Software Validation Process on the following pages.

Validating Data Repeater

There are two ways to validate (license) Data Repeater.

Single Machine License (using a validation code, locked to a single computer)

This method licenses your Data Repeater software without the use of a hardware key. This means that the validation code is tied to a single particular machine and the software can not be installed and validated on another machine without obtaining another validation code.

Hardware Key License (allows you to move the software to different PCs)

The Hardware Key is the preferred method since it allows greater flexibility when using Data Repeater because you can move the Hardware Key to any computer on which you wish to run the program. Also, a single BSI Hardware Key can be used to license all of BSI's software.

NOTE: If you have multiple user logins on your system, it is recommended that you register Data Repeater using the login that it will be running under.

Once you have **Data Repeater** installed and you have decided which license you require, follow the instructions in the next sections for either **Validating Using a USB Hardware Key** or **Validating Without Using a Hardware Key**, depending on your choice.

Validation Using a USB Hardware Key

NOTE: Hardware Keys require that you install the Sentinel Protection driver on your computer BEFORE inserting the Hardware Key. This driver is provided on the BSI Install CD. Please see the next section for important information.

Installing the Sentinel Driver

The **Sentinel USB** [Hardware Key] **Driver** is supplied on the BSI Install CD. Please ensure that you install this if you are using a Hardware Key to validate Data Repeater. There is no longer a need to perform further configuration of the Sentinel Driver as was the case in previous versions.

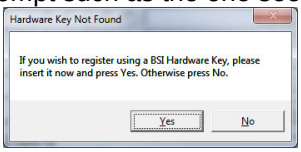
Once the driver has finished installing, plug the USB Hardware key into your system and wait for Windows to recognize it and then launch Data Repeater.

Validating Data Repeater

If you are using a hardware key to validate Data Repeater, make sure you have your USB Hardware Key inserted and that it's power light is on (you must have the Sentinel Driver installed before this step, see section above if you have not done so).

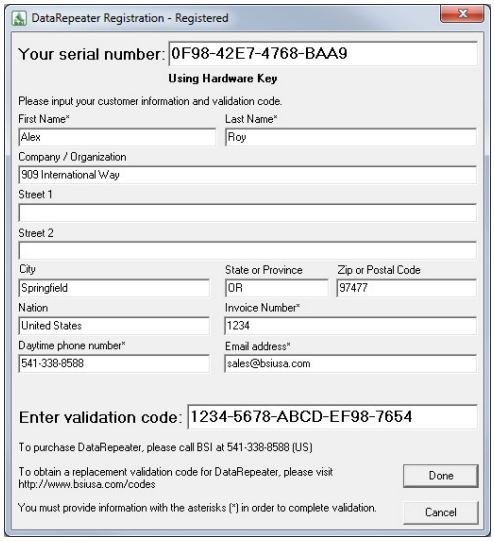
Once you have Data Repeater running, select **Help | Register** from the main Menu.

If you receive a prompt such as the one seen here:



…your Hardware Key was not detected by Data Repeater. Make sure the driver is properly installed and that the Hardware Key is fully plugged in and the power light is lit.

If you are using a Hardware Key, your **Serial Number** should match that shown on the tag attached to the Hardware Key. If not, check that the **Sentinel Driver** is installed correctly and that your hardware key is plugged in properly.



Enter all the information marked with an asterisk together with the Validation Code supplied with your dongle.

Keep a note of your Hardware Key Validation Code, you will need to re-enter this information if you move the Key to another computer.

Hardware Keys can also license other BSI products, so if you have a Hardware Key in your computer already, the license for Data Repeater can be added to the existing Hardware Key. Contact sales@bsiusa.com for pricing and order details.

Only **one** Hardware Key should be used in your computer at any time. You should not place two Hardware Keys in one machine at the same time. Contact BSI Sales if you need to install Data Repeater on a machine that already has a BSI Hardware Key installed.

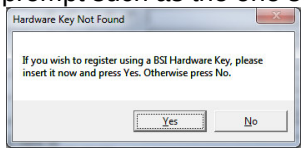
Validation Without a USB Hardware Key

To validate Data Repeater without a hardware key, you will need to install Data Repeater and run it. You will need to obtain the serial number generated by your system and then submit a validation code request via the www.bsiusa.com/codes website. The steps below will walk you through the process.

Validating Data Repeater

If you are NOT using a Hardware Key for validation of Data Repeater, open Data Repeater and select **Help | Register** from the main Menu.

You will receive a prompt such as the one seen here:

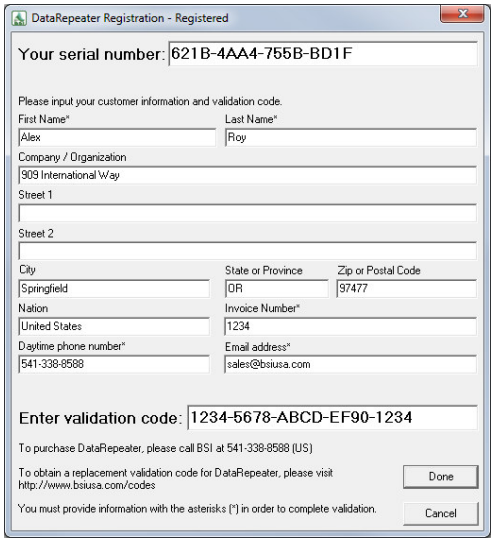


…press **No** to continue.

Your system generated **Serial Number** will be shown in the **Your Serial Number** field at the top of the **Registration** window as shown at right.

Copy this number down and proceed to www.bsiusa.com/codes. You will be prompted by the website to enter your Serial Number, Invoice Number, Company Name, Email Address, and perhaps some further information.

You will receive a return email with your validation code. Enter the validation code into the **Enter validation code** field at the bottom of the **Registration** window as shown at right.

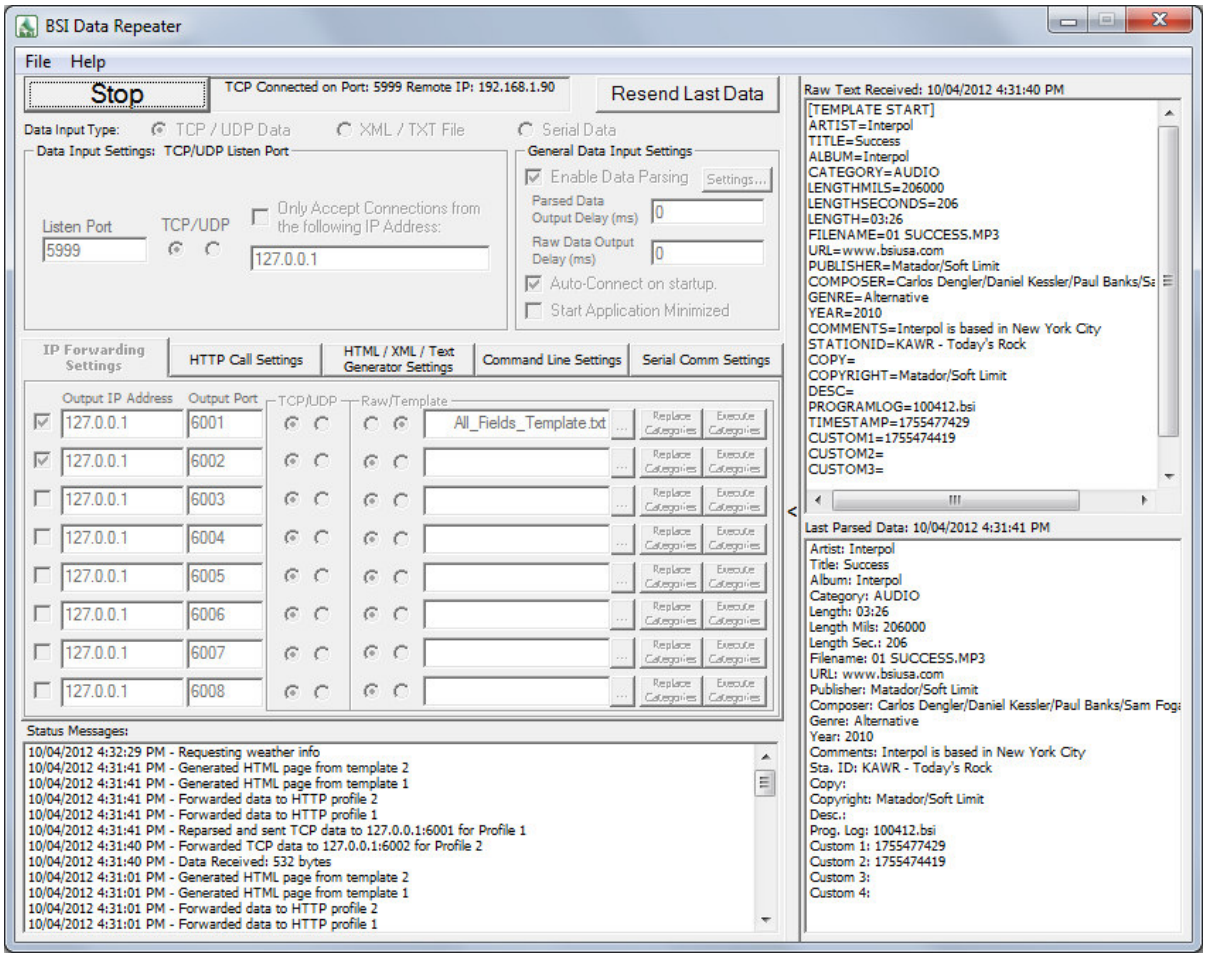


Once you have entered all the information and your **validation code**, click the **Done** button. Data Repeater will now be registered.

If you are not using a USB Hardware Key, please submit a validation code request by filling in the form at: http://www.bsiusa.com/codes

The User Interface

When you first start *Data Repeater* you will see the *Data Repeater* Main Screen as shown below.



Title Bar

Along the very top of the *Data Repeater* Main Screen you will see the *Data Repeater* title bar. When validated, the title bar will read "BSI Data Repeater". When not validated – and in demonstration mode – the title bar will read "BSI Data Repeater - UnRegistered". The right side of the title bar contains buttons to Minimize or Exit Data Repeater. Clicking on the Minimize button will hide the Data Repeater's main window and place an icon into the System Tray as shown at right. Clicking on the Exit button will completely close the application.

Menus

Below the title bar you will find the *Data Repeater* menu bar. To activate a menu, click on it with the left mouse button.

The Menu Bar contains menus for File and Help.

The **File** menu contains an option for **Exit**, which will completely close the Data Repeater application.

The **Help** menu contains the following options:

Help Contents – displays this manual in an "Online" help format
Register – which allows you to enter your validation code information
About – shows a pop-up window listing the version number

Main Controls

The top of the main user interface contains the **Start/Stop** button, which starts and stops the data listen and data forwarding functions.

To the right of the **Start/Stop** button is the status display. Information about the current state of the data listen function is displayed here.

To the right of the Status Display is the **Resend Last Data** button. This button will resend the last received data to all the data forwarding profiles you have configured. It is useful for troubleshooting connections to your data destinations. Note that this button will be "grayed out" until data is received via the selected data listen input type (TCP/UDP/Text File/Serial).

Data Input Type / Data Input Setup Settings

The **Data Input Type** selection dictates what data listen function Data Repeater will perform. Data Repeater can listen for data on your network using TCP or UDP protocols. It can also monitor a text based file (which can contain XML data, or tagged text) for new data (each time the text file is modified, Data Repeater will re-open and forward the data contained therein), and it can also listen on a Serial Port for data.

Depending on the **Data Input Type** selected, the **Data Input Settings** field will change to show pertinent settings for the selected input type. These settings dictate how Data Repeater will listen for data to forward to your data forwarding destinations.

General Data Input Settings

This section contains settings that apply to all data input types, including enabling and setup of Data Parsing, data output delay for raw data and parsed data, enabling Auto-Connection when Data Repeater starts up, and starting Data Repeater minimized.

IP Forwarding Settings

The 8 available profiles for your TCP/UDP data output are contained in this tab. Each profile contains settings for the destination IP address and port, the output type (TCP or UDP), and selections for simply forwarding the incoming data as it is received (Raw output), or re-parsing the incoming data and outputting the data using a user-defined template to reformat the data before it is output.

HTTP Call Settings

This output type is only available when Data Parsing has been configured and is enabled. Data Repeater will parse the incoming data and reformat it using each HTTP Call profiles' configuration before sending the HTTP Call to the URL/Port specified.

HTML / XML / Text Generator Settings

This output type is also only available when Data Parsing is configured and enabled. The HTML/XML/Text generator will parse the incoming data and reformat it using a template you create, and then save the reformatted file to your hard drive. The template and output are text based, so depending on how you format the contents of your template, you can have Data Repeater save an HTML, XML, or ASCII Text file each time new data is received.

Command Line Settings

The **Command Line** output profiles allow you to execute third party command line applications using Data Repeater. Command lines can contain meta-variables for artist, title, and more, so that you can include the data received in command line switches. This output type requires Data Parsing to be configured and enabled to be available.

Serial Comm. Settings

Serial data can be output by Data Repeater that can include artist, title, and more, by including meta-variables in the **Serial Output String** for each profile. You must have Data Parsing enabled and configured for this option to be available for use.

Status Message List

The **Status Messages** list will show incoming and outgoing data activity as it occurs as well as any errors encountered.

Raw Text Received List

The **Raw Text Received** list is shown when the main window is fully expanded (the right side of the window has a separator marked “<” when the window is expanded, or “>” when it has the right side fields hidden). This field displays the actual data received via the **Data Input Type** specified. Its title bar will also display the time the last data was received.

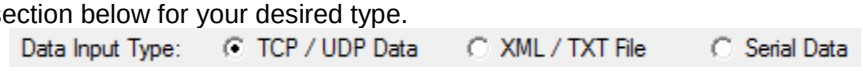
Last Parsed Data List

The **Last Parsed Data** list is shown when the main window is fully expanded (the right side of the window has a separator marked “<” when the window is expanded, or “>” when it has the right side fields hidden) below the **Raw Text Received** field. This list displays the fields Data Repeater was able to parse from the last incoming data. Its title bar will also display the time the data was parsed.

Setup

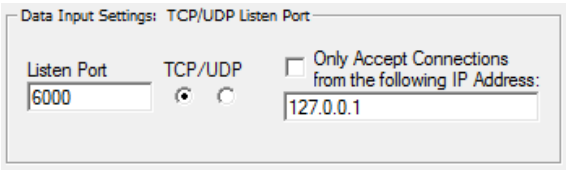
Configuring Data Input

The first step required to use Data Repeater is to configure your data input type. You will need to know what type of data you want to have Data Repeater receive: TCP, UDP, XML/Text File, or Serial (Data Repeater can only listen for one type of data input at a time). Once you have determined the input data type you want to use, select it from the **Data Input Type** selector (as shown below), you can then proceed to the section below for your desired type.



Configuring TCP/UDP data input

Selecting the **TCP/UDP Data** Input Type will cause the Data Input Settings field to display as shown at right. Type in the port number you want Data Repeater to listen into the **Listen Port** field. You will also need to select either TCP or UDP from the **TCP/UDP** selector, depending on what protocol you want.

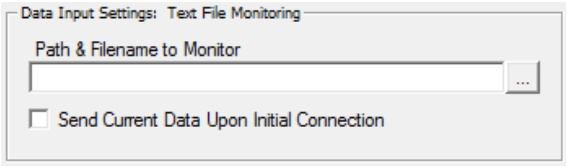


If you want to force Data Repeater to only accept data from a specified IP address (a good idea if your listen port is publicly available on the internet), enable the **Only Accept Connections from the following IP Address** check box and type in the IP address of the machine Data Repeater is supposed to be listening for.

Proceed to the Configuring General Data Input Settings section for instructions on completing the remaining data input settings.

Configuring XML/Text File data input

Selecting the **XML/TXT File** Input Type will cause the Data Input Settings field to display the fields at right. Using this option, Data Repeater will monitor the specified text file, and any time it detects that file has been changed/written to, it will re-open the file and read its contained text and use it as input data.



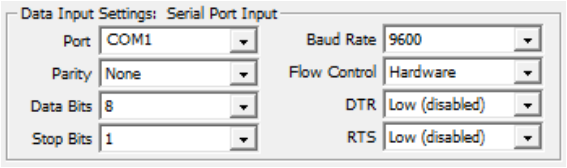
To select your desired text file, you can either type the full path and file name into the **Path & Filename to Monitor** field, or you can click on the ... button to browse to your file.

If you want Data Repeater to open the specified file as soon as data listening is started (rather than wait for the file to be updated), enable the **Send Current Data Upon Initial Connection** option.

Proceed to the Configuring General Data Input Settings section for instructions on completing the remaining data input settings.

Configuring Serial Data input

If you need Data Repeater to listen for data on a Serial Port, once you've selected the **Serial Data** Input Type, the fields shown at right will appear in the Data Input Settings field. Enter the required port settings as dictated by the device you are connecting to your serial port.



Proceed to the Configuring General Data Input Settings section for instructions on completing the remaining data input settings.

Configuring General Data Input Settings

The **General Data Input Settings** section contains settings applicable to all Data Input types. The available options are:

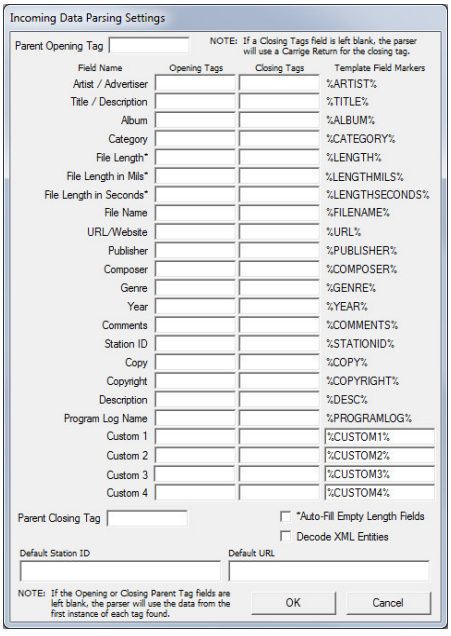
- **Enable Data Parsing** – Data parsing allows Data Repeater to determine specific entries for data such as artist, title, album, category, and more, so that the data output profiles that require can reformat the incoming data to customizable output formats. Without data parsing, it would not be possible to reformat the incoming data. See the Configuring Data Parsing Settings section below for instructions on setting up these settings.
- **Parsed Data Output Delay** – This field allows you to input a length of time in milliseconds that you would like the forwarding of your re-formatted (parsed) data to be delayed, or enter a 0 for no delay. This is useful for outputting meta-data to destinations that are using delayed audio, such as a profanity delay or a slow stream encoder.
- **Raw Data Output Delay** – This field performs a similar function to the above Parsed Data Output Delay function, but delays the forwarding of "raw data" (data that has not been re-parsed) to the TCP/UDP forwarding profiles when set to the Raw output option. The Raw data can be delayed by a different length than parsed data by entering the length of time in milliseconds that you desire, or not delayed at all by entering a 0.
- **Auto-Connect on startup** – With this option enabled, Data Repeater will immediately start the data listen function as soon as it is started up. When this option is disabled, the data listen function will not start until you manually click on the **Start** button after the application is started.
- **Start Application Minimized** – By default, Data Repeater will be visible on your screen when it is opened. If you want Data Repeater to open minimized, leaving just its icon in the system tray (the area next to the clock that shows background application icons), enable this option. When Data Repeater is minimized to the system tray, all you have to do is double-click on its icon to maximize it and make it visible.

Configuring Data Parsing Settings

Data parsing is the function that allows Data Repeater to reformat your incoming data before forwarding it to the data output locations. The Data Parsing settings help Data Repeater determine where in your incoming data it should look for particular pieces of data. If the Data Parsing settings are incorrect or are inconsistent in your incoming data, it is not possible to properly reparse your incoming data, so be very aware of how your data is formatted so the settings can be entered correctly.

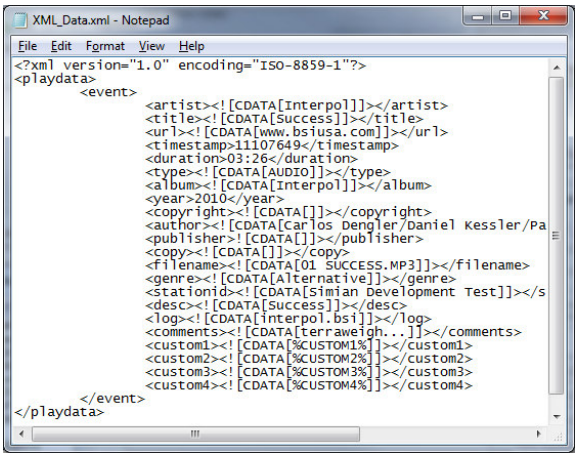
If you do not have an example of the input data you want Data Repeater to use, you can configure Data Repeater to receive your data, start the data input listen function by clicking on the **Start** button, then look at the data received by Data Repeater.

The image at right is the **Incoming Data Parsing Settings** window. There are quite a few settings required in this window, so you might guess that setting up data parsing is pretty complex, but in reality it is pretty simple. Each line in this window is how Data Repeater finds the start and end of each field's data, the majority of lines in this window are a repeat of the same start/end setting for each field.



Configuring for XML Data Input

XML formatted data is very easy to set up in the Data Parsing Settings, since it is at its heart a standard created exactly for saving data that is easily parsed. At right is an example of XML formatted data:



If your data does not appear similarly to the document shown at right, you may have standard text formatted data. If that is the case, proceed to the Configuring for Text Data section later in this manual.

Notice that the main portion of the document's data we care about is surrounded by opening and closing tags, represented by the <event> and </event> "tags". This is referred to as a "node" in XML parlance. There may be other nodes in the data you receive that contain duplicate artist, title, album, or other fields, so it is important to determine which "node" designation is the one unique to the data you want Data Repeater to use. In this case, you'd want to enter "<event>" into the **Parent Opening Tag** field, and "</event>" into the **Parent Closing Tag** field. With that information, Data Repeater will only pay attention to the data within the main audio event node (the example document only has a single "event" node, but yours may have others).

First, let's look at the artist data. The artist contained in this document is "Interpol", and the text directly preceding "Interpol" is "<artist><!CDATA[". This means something specific in XML, the explanation of which is outside the scope of this manual, but needless to say, this text always appears before the artist name in this instance (You can watch the input data for several artists to check what your tag is. Even if the opening artist tag in your data input is slightly different, there should always be the same "chunk" of text before your artist name, no matter which artist is listed, that "chunk" is what is important here). Because the artist name is always preceded by that text, "<artist><!CDATA[" is what we will want to enter into the **Artist / Advertiser Opening Tag** field. Also notice that the artist name is followed by "]]></artist>", this is the unique identifier that signifies the end of the artist data, so we want to enter "]]></artist>" into the **Artist / Advertiser Closing Tag** field. With these two pieces of data, Data Repeater can now figure out who the artist is in the data received!

Using the steps in the above paragraph, you can now determine the tags that denote the start and end of the Title field, URL field, Album field, and more. Don't worry if you don't have all the available fields in your incoming data. Simply leave the opening and closing tag fields blank for any fields you don't have in your incoming data.

If your incoming data only has one or two of the available length fields, you can enable the ***Auto-Fill Empty Length Fields** option, and Data Repeater will compute the remaining length fields based on the length data you do have available for the fields you don't have.

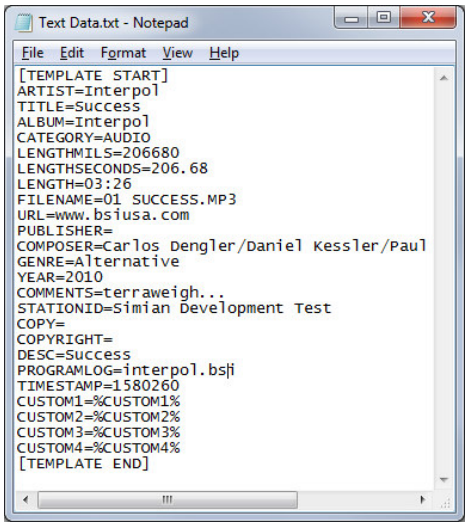
The XML format has the potential to use what are referred to as "escape characters": characters, such as "&" are not valid in some XML tags, so they are converted to equivalent codes (& in the case of "&"). If your incoming data contains these escape characters, make sure to enable the **Decode XML Entities** so these escape characters are converted back to their normal characters in the parsed data.

The **Default Station ID** and **Default URL** fields allow you to use a default entry of your choosing if your incoming data does not contain an entry for either of these fields. If the incoming data is able to parse entries for these fields, that data will be used instead.

Once you have all of your Data Parsing Settings entered, click **OK** to save your changes. You can try out your settings to make sure Data Repeater is properly parsing your incoming data by watching the **Last Parsed Data** field as Data Repeater receives incoming data. It lists each available field and what data has been parsed for each field. Once all your fields are being parsed properly, you are ready to proceed to the Configuring Data Output Profiles section later in this manual to configure some data output profiles.

Configuring for Text Data Input

If you have determined that your incoming data is not in XML format, you may still be able to use the data. For the data to be usable by Data Repeater, each field must have some sort of indication of what the field is (also signifying the start of the field's data), and it must either have a carriage return at the end of each field, or another text indication of the end of the field.



Ideally, the entire document would have an opening tag (in the case shown at right, that is the text "[TEMPLATE START]") and a closing tag ("[TEMPLATE END]" in this case). As long as there aren't repeating entries for artist, title, album, etc., these opening and closing "Parent" tags aren't necessary. If your data does contain Parent tags, you can enter the top tag into the **Parent Opening Tag** field, and the bottom tag into the **Parent Closing Tag** field in the **Incoming Data Parsing Settings** window to aid in the parsing process, otherwise an entry is not strictly necessary.

In the example text data shown at right, each field is preceded by the field name and an equal symbol. The end of each line has a carriage return (new line) character. With both of these pieces of information, Data Repeater will have enough information to determine the start and end of each field, and with each fields' unique field name identifier it will able to tell what each fields' data is.

Let's look at the Artist field. In this case, the current artist is "Interpol". With this info, we can tell that the Artist field is preceded by the text "ARTIST=", and since there is a new line after the text "Interpol", we know that there is a carriage return after the artist field data. That means that we can enter "ARTIST=" into the **Artist / Advertiser Opening Tag** field. That leaves the **Artist / Advertiser Closing Tag** field needing an entry. The Closing Tag fields will automatically search for a carriage return as the closing tag for all empty entries, so in this case we will leave the **Artist / Advertiser Closing Tag** field empty. If your incoming data *does* include text after each fields' data, enter it in the corresponding **Closing Tag** field.

Now the process just needs to be repeated for the Title, Album, Category, and other remaining fields.

If your incoming data only has one or two of the available length fields, you can enable the ***Auto-Fill Empty Length Fields** option, and Data Repeater will compute the remaining length fields you don't have data for, based on the length data you do have available.

Non-XML format data usually will not have what are referred to as "escape characters" (characters, such as "&" are not valid in some XML tags for instance, so they are converted to equivalent codes: & in the case of "&"). If your incoming data contains these escape characters, make sure to enable the **Decode XML Entities** so these escape characters are converted back to their normal characters in the parsed data. If you don't see any escape characters in your data, leave this option unchecked.

The **Default Station ID** and **Default URL** fields allow you to use a default entry of your choosing if your incoming data does not contain an entry for either of these fields. If the incoming data does contain parsable entries for these fields, that data will be used instead.

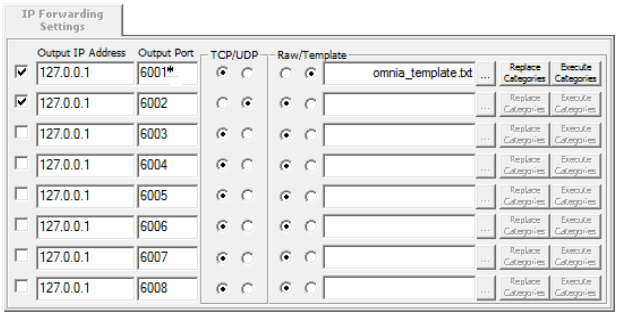
Once you have all of your Data Parsing Settings entered, click **OK** to save your changes. You can try out your settings to make sure Data Repeater is properly parsing your incoming data by watching the **Last Parsed Data** field as Data Repeater receives incoming data. It lists each available field and what data has been parsed for each field. Once all your fields are being parsed properly, you are ready to proceed to the Configuring Data Output Profiles section below to configure some data output profiles.

Configuring Data Output Profiles

The previous sections of this manual are all about getting data *into* Data Repeater. The magic comes when you configure the data output profiles, which forward the received data to other destinations, either exactly as received (Raw TCP/UDP output), or reformatting the received data using your own user-created templates and then sending the data out via TCP/UDP, HTTP Call, HTML/XML/Text, Serial, and/or executing a command line application. The following sections walk you through setting up all the available data output profiles, all of which can output simultaneously.

Setting Up an IP (TCP/UDP) Output Profile

To forward your data to a TCP or UDP destination, you will use one of the **IP Forwarding** profiles. As shown at right, there are 8 available profiles. To configure a profile, you will need the IP address and port for your destination device/computer, and you will need to know if that destination will be using TCP or UDP protocol. Enter this information into the **Output IP Address**, **Output Port** fields and select the protocol from the **TCP/UDP** selector.



DataRepeater normally keeps TCP connections open once they are established. If your destination application expects the TCP connection to be closed after each data output, add an asterisk (*) after the port number (as shown in the picture above). This signifies to Data Repeater that needs to close the connection after each data set has been delivered to the destination application.

If the incoming data format is the exact format your destination device expects, you can choose **Raw** from the **Raw/Template** selector. The Incoming Data Parsing function does not need to be enabled or configured to use the Raw output option.

On the other hand, if your destination device needs the data to be reformatted so it'll understand the data, you'll need to create a TCP/UDP Output Template. The next section of this manual, titled Creating a TCP/UDP Output Template walks you through that process. Once you have your template created, select the **Template** option from the **Raw/Template** selector, and choose your template file by clicking on the ... button and browsing for your template file. Note that the Incoming Data Parsing function must be enabled and configured to be able to use the template functionality.

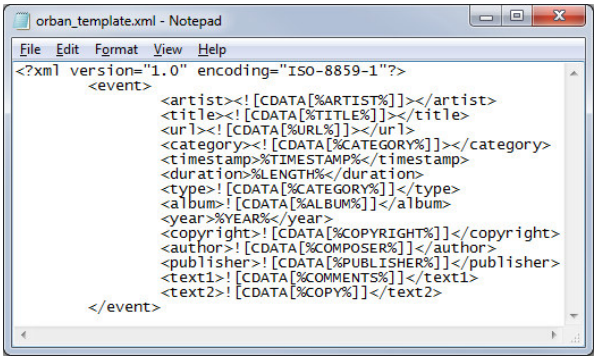
To enable your IP Forwarding profile, place a check mark in the check box to the left of the corresponding **Output IP Address** field. If you ever want to disable one of the profiles while keeping your settings, just uncheck the box and Data Repeater will ignore that profile.

Each IP Forwarding profile contains optional **Replace Categories** and **Execute Categories** functions. For more information on either of these options and how to configure them, see the sections titled Setting Up Replace Categories and Setting Up Execute Categories.

Creating a TCP/UDP Output Template

Data Repeater uses templates for reformatting input data. To create a TCP/UDP Output template, you will need to use an ASCII text editor, such as notepad.exe, to create your template. Because templates are made using a text editor, you can create your template in any formatting style you want: XML, INI, etc..

The example at right shows an XML formatted template. If you are unfamiliar with XML, it is simply a way of enclosing each piece of data within opening and closing tags so it is easy for computers to understand.



The format you will need to use is dictated by what ever device you are sending data to. You should be able to find/get an example of the data format your device requires from that device's manufacturer.

Notice that the template contains meta variables. These are %ARTIST%, %TITLE%, and other tags. The meta variables are used as keywords that Data Repeater will replace when it uses the template for reformatting the data. All the other text in the template than the meta variables will remain. The final result is the data format you created, and the meta variables will be removed and replaced with the corresponding field data received by the incoming data's Data Parsing function.

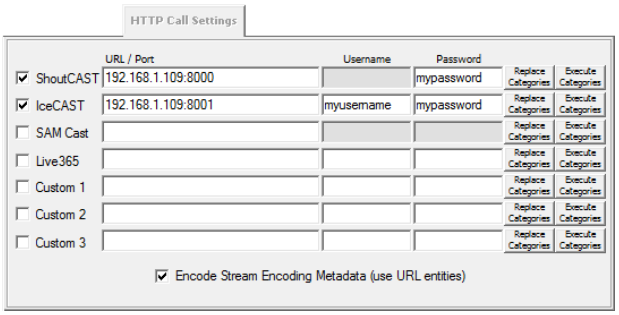
To create your custom template, paste your destination device's example format (or manually type it in) to a new text document in your ASCII text editor (again, notepad.exe is the easiest, most common text editor around), but replace the example's artist, title, category, etc, with the meta-variable field markers. The available meta variables are as follows:

- %ARTIST%
- %TITLE%
- %ALBUM%
- %CATEGORY%
- %LENGTH%
- %LENGTHMILS%
- %LENGTHSECONDS%
- %FILENAME%
- %URL%
- %PUBLISHER%
- %COMPOSER%
- %GENRE%
- %YEAR%
- %COMMENTS%
- %STATIONID%
- %COPY%
- %DESC%
- %PROGRAMLOG%
- %CUSTOM1%
- %CUSTOM2%
- %CUSTOM3%
- %CUSTOM4%

Once you have completed your template, save it to a location on your hard drive, such as C:\BSI32\ (the default installation location of Data Repeater). Your template is now ready for use in a TCP/UDP output profile in Data Repeater.

Setting Up an HTTP Call Output Profile

HTTP Call output is used by many internet stream encoders as an easy means of sending data from automation systems to your stream's listeners.



HTTP Call output requires that you have Input Data Parsing configured and enabled, so make sure that is all set up before proceeding.

As shown at right, there are 7 available HTTP Call output profiles. The check box to the left of each profile enables or disables that profile, so if you want to disable a profile without destroying that profile's settings, simply uncheck the check box and Data Repeater will ignore that profile.

Also note that there are four pre-configured profiles for ShoutCAST, IceCAST, SAM Cast (previously known as SimpleCAST), and Live365. These profiles are the most common stream encoders BSI runs into, so we set them up for you. There are also three custom profiles available. Note that all of these profiles are actually customizable if need be, but the pre-configured profiles output differing formats for the beginning of the URL (the part inserted before the section that comes from the editable profile data). See the next section, titled Editing an HTTP Call Profile's Output Template, for the steps required to edit any of the available HTTP Call output formats.

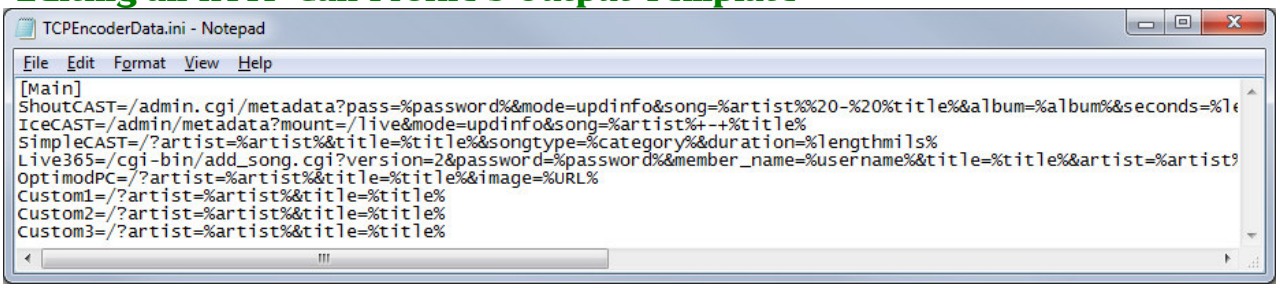
To configure your HTTP Call profile of choice, you will need to enter the URL (or IP address) and port number, separated by a colon, into the **URL/Port** field, and also enter the username and/or password for the profiles that require them, into the **Username** and **Password** fields.

Each HTTP Call profile contains optional **Replace Categories** and **Execute Categories** functions. For more information on either of these options and how to configure them, see the sections titled Setting Up Replace Categories and Setting Up Execute Categories.

Note that the ShoutCAST profile behaves a little different than the other profiles. ShoutCAST requires special category letter codes, rather than full categories, so when using the ShoutCAST profile, Data Repeater will automatically convert several common standard category names to the equivalent ShoutCAST category letter code for you. Keep this in mind if you are trying to re-purpose the ShoutCAST output profile for other purposes. The category conversions are as follows:

Category Name as entered in Simian	Character replacement before sending to SimpleCast
"Audio"	S
"Ad", "Advertisement", "Comm", "Commercial"	A
"ID", "Station ID"	I
"Promo", "PSA"	P
"Jingle"	J

Editing an HTTP Call Profile's Output Template



To edit one of the available HTTP Call profile output templates, you will need to open the file **TCPEncoderData.ini** in an ASCII text editor, such as notepad.exe. TCPEncoderData.ini is located in Data Repeater's installation directory, which is C:\BSI32\ in a default installation.

The image above shows the contents of TCPEncoderData.ini, each line starts with the HTTP Call profile's name, followed by an equal sign, and then the profile's output template.

Each output template contains the required text as dictated by the receiving encoder's required format, but notice the meta variables contained within each template's text. When Data Repeater reformats the incoming data to be output, it replaces the meta variables with the corresponding data parsed from the incoming data, so %ARTIST% for example would be replaced by the actual artist name received, then the reformatted data gets sent to the HTTP Call destination. All text that is not a meta variable is used as is in the HTTP Call output's reformatting process. Also note that the template does not start with a URL or Port, this information is taken from the HTTP Call's **URL/Port** field, and added to the beginning of the template text from here in the TCPEncoderData.ini file when output.

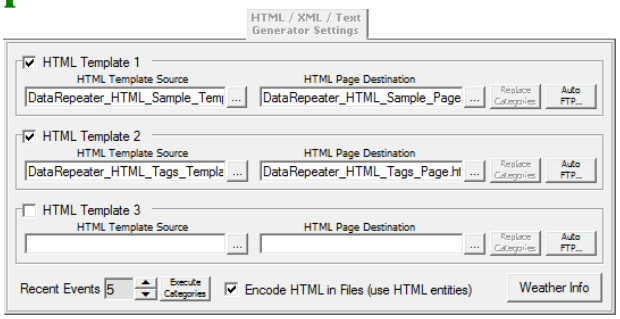
To edit a template, simply enter the text modifications you require, but rather than including actual artist names and titles, etc., here, use meta variables instead. The available meta variables are as follows:

- %ARTIST%
- %TITLE%
- %ALBUM%
- %CATEGORY%
- %LENGTH%
- %LENGTHMILS%
- %LENGTHSECONDS%
- %FILENAME%
- %URL%
- %PUBLISHER%
- %COMPOSER%
- %GENRE%
- %YEAR%
- %COMMENTS%
- %STATIONID%
- %COPY%
- %DESC%
- %PROGRAMLOG%
- %CUSTOM1%
- %CUSTOM2%
- %CUSTOM3%
- %CUSTOM4%
- %PASSWORD%
- %USERNAME%

Note that the %PASSWORD% and %USERNAME% meta variables will be replaced with the corresponding username and password entered into the **Username** and **Password** fields in each profiles' configuration settings.

Setting Up an HTML/XML/Text Output Profile

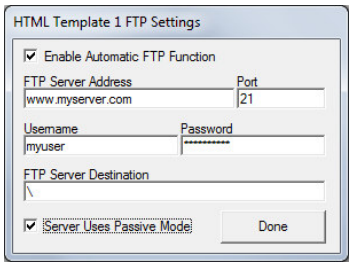
Each HTML/XML/Text Generator profile outputs an ASCII Text based file to the path specified in the **HTML Page Destination** field, using the specified HTML output template to reformat the incoming data. Note that even though the labels specify the template and destination as being HTML, this is not strictly true (hence the tab being called the HTML/XML/Text Generator Settings). HTML is a text based format, so your template can actually be in any ASCII text based format you want, including XML or a simple text document, and the output will match the template.



Each time the HTML Generator creates a document, it opens the specified template file, replaces the HTML meta variables with the corresponding input data's parsed fields, and then it saves the reformatted document to the path and filename specified in the **HTML Page Destination** field.

For instructions on creating your own HTML/XML/Text output template, see the next section in this manual titled Creating an HTML/XML/Text Output Template.

If you require the output file to be copied across the internet to your web server, you can enable the **Auto FTP…** options by clicking on the **Auto FTP…** button. The window shown at right will appear. Enter the settings your FTP server requires in the available fields, then place a check in the checkbox for **Enable Automatic FTP Function** and click **Done**.

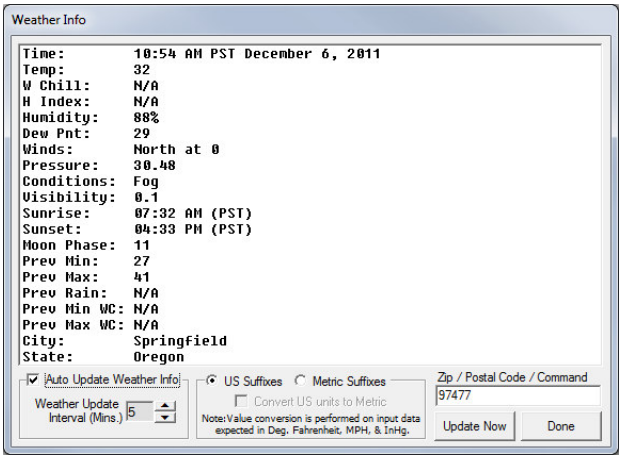


Each HTML/XML/Text output profile contains an optional **Execute Categories** function (note that they do not contain Replace Categories). For more information on this optional function and how to configure it, see the section titled Setting Up Execute Categories.

The HTML Generator also has fields that list the most recently played audio files. This function adds the parsed data from the incoming data to a list that keeps track of up to the last 99 files played (the default is 5 most recent items). You can select the number of items you want Data Repeater to remember by adjusting the number listed in the **Recent Events** selector. Also note that the Recent Events function has an Execute Categories option, which allows you to exclude or include only certain categories of files in the recent list.

In most cases, if you are generating HTML files using the HTML Generator (as opposed to XML or Text files), you will want to enable the **Encode HTML in Files (use HTML entities)** option so that incompatible characters are replaced by their HTML "escape character" equivalent.

The HTML Generator also has current weather fields available to it. The weather information used in these fields are obtained using an API available through www.wunderground.com. To configure the weather retrieval settings, click on the **Weather Info** button and the window shown at right will appear.



If you are in the US or have a compatible international postal code, enter your zip/postal code into the **Zip/Postal Code/Command** field.

If you want to enter an airport, IATA, or ICAO code rather than a zip code, you can enter the keyword "STATION", separated by a space, and then your desired airport code ("STATION EUG" for example)

into the **Zip/Postal Code/Command** field.

You can also have Data Repeater read weather data from an INI file by entering the keyword "FILENAME", separated by a space from the path to your .ini file in the **Zip/Postal Code/Command** field. For more information on setting up .ini file based weather reading, contact BSI Tech Support via http://support.bsiusa.com.

You can have Data Repeater automatically retrieve the weather information by placing a check mark in the **Auto Update Weather Info** field and set the interval at which Data Repeater will retrieve the weather info by adjusting the **Weather Update Interval (Mins.)** selector. If you are not using any of the weather info in any of your HTML output templates, it is recommend to disable this feature so as not to use unnecessary processor time on your computer.

Data Repeater can also display metric unit designations and/or convert the weather info from US units into metric units by setting the unit selector to the **Metric Suffixes** option and enabling the **Convert US units to Metric** check box.

The Data Repeater includes a couple demonstration HTML templates for you to try out. They are located in Data Repeaters install directory (C:\BSI32\, by default). You can try out the HTML Generator with these templates to see what it will do.

Make sure to name your HTML Destination Page file name differently from the Template Source file name so that Data Repeater will not overwrite your template file with a merged output copy of your page.

To create an HTML/XML/Text output template of your own, see the section below in this manual titled Creating an HTML/XML/Text Output Template.

The first sample HTML template is named **TCPRepeater_HTML_Sample_Template.htm**. It is a sample page that you might use as your stations "currently playing" page.

The second sample HTML template is named **TCPRepeater_HTML_Tags_Template.htm**. It is a demonstration page that lists all of the available HTML meta variable markers and what their output is for the currently parsed input data and weather info currently retrieved. You can use this template to quickly look up what meta variables are available to you to put into your own HTML template.

Creating an HTML/XML/Text Output Template

Specific instructions on creating HTML and/or XML pages are both beyond the scope of this manual, so if you are unfamiliar with creating either of these types of documents you will need to find some literature on how to create these types of documents, or you will need to have a competent web developer do it for you.

With the following steps, you will be able to create templates that match your website's layout and/or match your needed XML or text file output format. You can also use the HTML Generator to create text based files if your desired destination for your data is a text file on your hard drive (some RDS or stream encoders can use a file rather than direct TCP or HTTP data reception to get your audio data).

To create a template, you will need to use a text editor or HTML editor. When you create your template, rather than using an actual artist name or song title (or other field data), you will use "field markers" (AKA meta variables). When Data Repeater receives input data and then outputs your HTML/XML/Text document, it opens your HTML template, searches for and replaces a specific list of field markers with data obtained from the data input's Data Parsing function and Weather Info function, then saves that merged copy to the destination location specified in your HTML Output profiles' settings.

The meta variable field markers available to the HTML Generator are a little different than the meta variables available to the other data output functions. This is because they need to conform to the HTML comment structure. The HTML meta variables available are as follows:

<!--BSIARTIST-->	The artist/advertiser from the current input data
<!--BSITITLE-->	The title/description from the current input data
<!--BSICURRENT-->	The play time, artist, and title from the current input data
<!--BSICURRENT1-->	The artist and title from the current input data
<!--BSICURRENTFILE-->	The filename from the current input data
<!--BSIALBUM-->	The album name from the current input data
<!--BSICOPYRIGHT-->	The copyright from the current input data
<!--BSICOMMENTS-->	The comments from the current input data
<!--BSICOMPOSER-->	The composer from the current input data
<!--BSIYEAR-->	The year from the current input data
<!--BSIGENRE-->	The genre from the current input data
<!--BSIPUBLISHER-->	The publisher from the current input data
<!--BSIURL-->	The URL/Website from the current input data
<!--BSIALBUMARTFILENAME-->	The album art file name from the current input data*
<!--BSIRECENT-->	A long format list of all the recent items played
<!--BSIRECENTX1--> through <!--BSIRECENTX##-->	Lists an individual recent item's play time, artist name title, and year (where ## is the specific recent item's number, without leading zero)
<!--BSIRECENT1-->	A short format list of all the recent items played
<!--BSIRECNET1X1--> through <!--BSIRECENT1X1##-->	Lists an individual recent item's artist name and title (where ## is the specific recent item's number, without leading zero)
<!--BSIRECENTFILES-->	A list of the filenames of the recent items played

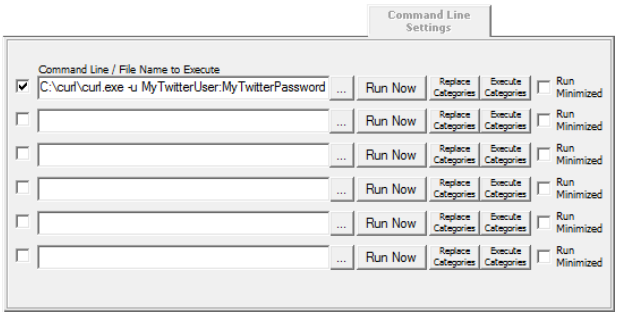
• <!--BSIRECENTFILES1--> through <!--BSIRECENTFILES##-->	Lists an individual recent item's file name (where ## is the specific item's number, without leading zero)
• <!--BSIWEATHER-->	A multi-field listing of the current weather information
• <!--BSIWEATHERAVAILABLE-->	Displays Yes/No if weather info has been retrieved
• <!--BSIWEATHERTIME-->	Time weather info was updated by wunderground.com
• <!--BSIWEATHERCITY-->	City weather info is for
• <!--BSIWEATHERSTATE-->	State weather info is for
• <!--BSIWEATHERTEMPERATURE-->	Current temperature
• <!--BSIWEATHERCONDITIONS-->	Current sky conditions (cloudy, sunny, calm, etc.)
• <!--BSIWEATHERWINDS-->	Current Wind direction and speed
• <!--BSIWEATHERPRESSURE-->	Current barometric pressure of the atmosphere
• <!--BSIWEATHERHUMIDITY-->	Current percent humidity
• <!--BSIWEATHERVISIBILITY-->	Current visibility distance and conditions
• <!--BSISTATIONID-->	Station ID text (from input data, or from default entry)
• <!--BSITIME-->	System time on the computer running Data Repeater
• <!--BSIDATE-->	System date on the computer running Data Repeater
• <!--BSICOMINGUP-->	N/A**
• <!--BSICOMINGUP1--> through <!--BSICOMINGUP##-->	N/A**
• <!--BSICOMINGUPFILES-->	N/A**
• <!--BSICOMINGUPFILES1--> through <!--BSICOMINGUPFILES1##-->	N/A**

*The <!--ALBUMARTFILENAME--> tag gets a file name based on the artist and album names obtained from the input data's parsed fields. The HTML Generator searches within the \Album Art\ folder in Data Repeater's install directory for .jpg, .gif, .bmp, and .png files that match the formats [artist] - [album].[jpg/gif/bmp/png], [album].[jpg/gif/bmp/png], or [artist]. [jpg/gif/bmp/png], if it finds one of these files, it uses the file name found as the album art file name. This way, you can have a corresponding album art folder on your web server and reference album art images in your html page to display the image name found. If an album art image is not found that matches the above criteria, the tag defaults to "default.jpg", so you can place your station's logo onto your local \Album Art\ folder and corresponding album art folder on your web server and name it default.jpg to display that logo on your web page instead.

**Note: some unsupported fields are listed, since this feature supports templates created with Simian's HTML Generator. For instance, Data Repeater has no way of preemptively knowing what audio files are coming up, so these field markers are left blank when used with Data Repeater's HTML Generator.

Setting Up a Command Line Execute Profile

If you have any third party applications that are executable via command line that you want to execute each time incoming data is received, enter its command line path into one of the available **Command Line / File Name to Execute** fields and enable it by placing a check mark in the check box to the left of that profile.



If the command line app accepts command line switches for artist, title, or any other available field, you can include meta-variables within your command. The meta-variables available are:

- %ARTIST%
- %TITLE%
- %ALBUM%
- %CATEGORY%
- %LENGTH%
- %LENGTHMILS%
- %LENGTHSECONDS%
- %FILENAME%
- %URL%
- %PUBLISHER%
- %COMPOSER%
- %GENRE%
- %YEAR%
- %COMMENTS%
- %STATIONID%
- %COPY%
- %DESC%
- %PROGRAMLOG%
- %CUSTOM1%
- %CUSTOM2%
- %CUSTOM3%
- %CUSTOM4%

The **Run Now** button to the right of each command line profile allows you to execute the specified command line manually (usually for testing purposes).

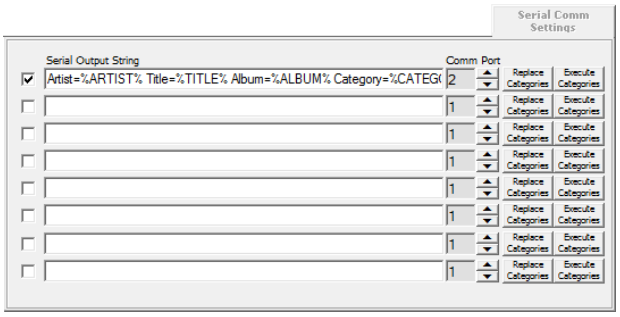
Each Command Line profile contains optional **Replace Categories** and **Execute Categories** functions. For more information on either of these options and how to configure them, see the sections titled Setting Up Replace Categories and Setting Up Execute Categories.

The example image above contains the command line:

```
C:\curl\curl.exe -u MyTwitterUser:MyTwitterPassword -d "status=The currently playing song is %TITLE% by %ARTIST%" http://api.supertweet.net/1/statuses/update.xml
```

This updates a Twitter© account each time new incoming data is received using “curl.exe” and an account set up via www.supertweet.net. Each time the command line is run, the %TITLE% meta-variable is replaced by the parsed title obtained from the incoming data, along with %ARTIST% being replaced by the parsed artist data. That means that each time incoming data is received, the command line is executed, but the artist and title meta variables have the actual data of the last received data instead.

Setting Up a Serial Comm. Output Profile



Data Repeater can output your incoming data through your serial ports (comm. ports 1-8 are supported) on your computer. Data Parsing must be enabled and configured to be able to utilize Serial Output.

To enable a Serial Output profile, place a check mark in the check box to the left of your desired profile.

Enter the text you want to be sent out the serial port into the **Serial Output String** field of your desired profile, but instead of entering an actual artist, title, etc. into the text string, use meta variables. When Data Repeater is outputting your serial string, it searches through the string for any meta variables, and when it finds them, it replaces them with the corresponding data obtained from the input data's Parse Data fields. The example in the image above would output "Artist=Interpol Title=Success Album=Interpol Category=AUDIO" for the song "Success" by the artist "Interpol".

The list of meta variable fields available to the Serial Output function are as follows:

- %ARTIST%
- %TITLE%
- %ALBUM%
- %CATEGORY%
- %LENGTH%
- %LENGTHMILS%
- %LENGTHSECONDS%
- %FILENAME%
- %URL%
- %PUBLISHER%
- %COMPOSER%
- %GENRE%
- %YEAR%
- %COMMENTS%
- %STATIONID%
- %COPY%
- %DESC%
- %PROGRAMLOG%
- %CUSTOM1%
- %CUSTOM2%
- %CUSTOM3%
- %CUSTOM4%

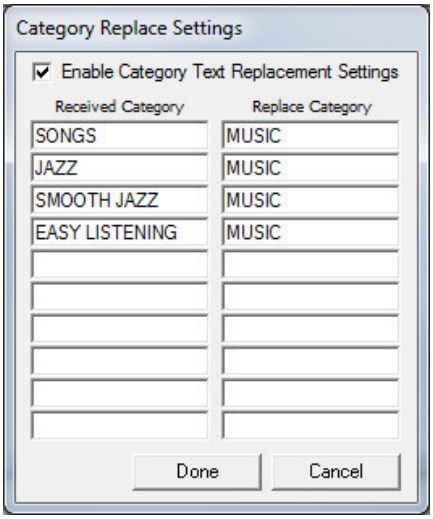
Select your desired output serial comm. port number from the **Comm. Port** selector. The selector allows you to select comm. ports 1 through 8, even if your system doesn't have 8 ports available, so make sure you select the correct port number as dictated by your Windows Port settings.

Each Serial Data output profile contains optional **Replace Categories** and **Execute Categories** functions. For more information on either of these options and how to configure them, see the sections titled Setting Up Replace Categories and Setting Up Execute Categories.

Setting Up Replace Categories

Replace Categories allow you to have Data Repeater output an alternate category name or code when specific category text is received from the input data. Many of the data output profiles have optional Replace Categories (all but the HTML generator has this option available).

A button for **Replace Categories** will appear within the settings for the profiles that do support them, click on it to access the Replace Category settings for your desired output data profile. You will be presented with the window shown at right.



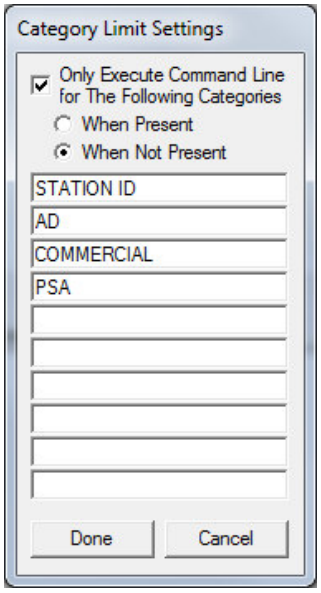
If the **Enable Category Text Replacement Settings** check box is enabled and the current parsed category text is listed in the **Received Category** column on the left, the %CATEGORY% meta-variable for the associated data output profile will use the text from the **Replace Category** column instead of the parsed category.

If you want Data Repeater to replace an empty received category with text, enter "<BLANK>" into the **Received Category** field and enter the category name you want it to be replaced with in the **Replace Category** field. On the other hand, if you want a specific received category to be replaced with a blank category, enter the desired category name into the **Received Category**, but leave the **Replace Category** field blank.

This is useful if the destination application of the data output requires a different category/code than the data you receive from the data input.

Setting Up Execute Categories

If you want no data output to occur for certain categories, click on the **Execute Categories** button next to your desired data output profile. A window similar to the one shown at right will pop up.



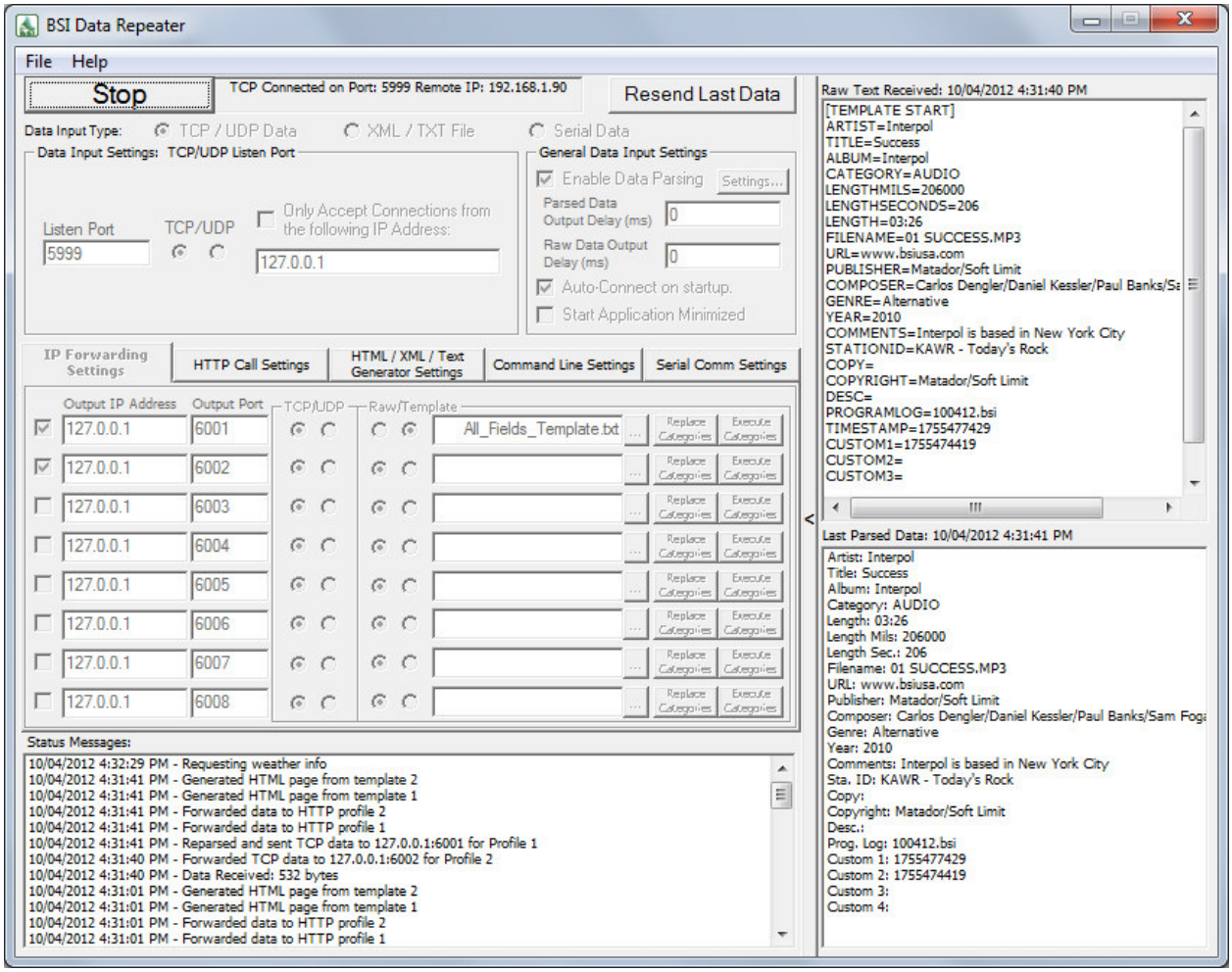
To enable the Execute Categories function, place a check mark into the **Only Execute Command Line for The Following Categories** check box.

Execute Categories can function in two different modes, either the data output profile will be executed whenever the currently parsed category name *is* in the list, or it will be executed whenever that category is *not* one of the categories in the list. Choose your desired mode by either selecting the **When Present** option, or the **When Not Present** option.

Enter your desired categories into the 10 available category fields (or, if you want to use an empty category entry, type in "<BLANK>", without quotes, into the field).

The Replace Categories function occurs before the Exclude Categories function takes place. This means that the categories referenced in the Exclude Categories list need to take that into account. If you are replacing the "Music" category with "Songs", and you want to forward data only when the "Music" category is seen, you would need to list "Songs" in the Exclude Categories rather than "Music" since "Music" would already have been replaced with "Songs".

Operation



Once you have configured your input data settings, and any desired data output profiles, general operation of Data Repeater is really quite simple. To start Data Repeater's listen function, click on the **Start** button at the top of the user interface. The Start button will subsequently change to be the **Stop** button, which when clicked will stop the Data Repeater's listen function and take it off line.

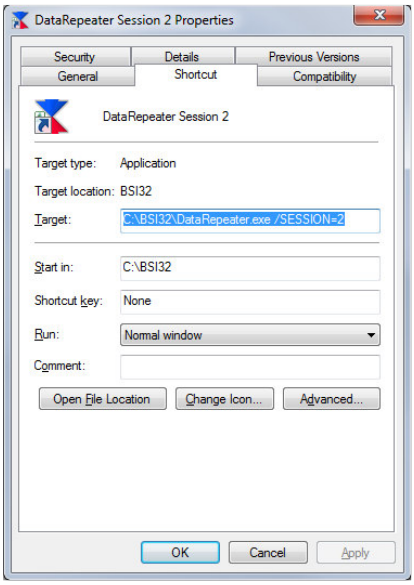
If you want Data Repeater to start listening for data as soon as it is started, just place a check mark into the **Auto-Connect on Startup** check box within the **General Data Input Settings** section shown above. With this option checked, the next time you start Data Repeater it'll start listening, just as if you clicked on the **Start** button.

If you want Data Repeater to run "in the background" when it is started up, without opening up the user interface, place a check mark into the **Start Application Minimized** check box in the **General Data Input Settings** section shown above. The next time you start Data Repeater it'll just go straight to the "System Tray" (the location on Window's Start Bar that contains the system clock and icons for your currently running background applications) as shown in the image above, pointed to by the red arrow and circled in red. To open or re-open Data Repeater when it is minimized to the system tray, just double click on it and the main user interface will open up on your screen.

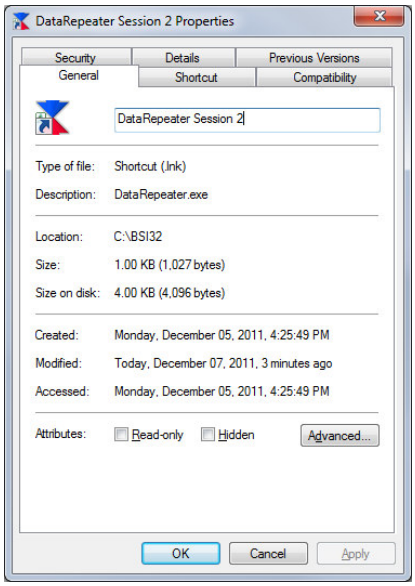
Using Multiple Instances Of Data Repeater At The Same Time

It is possible to open multiple instances of Data Repeater at the same time and have each instance retain its own program settings. You will need to create individual shortcut icons for each unique instance of Data Repeater you want to use. To do this, use the following steps:

1. Browse to your DataRepeater.exe file, located in C:\BSI32 on a default installation.
2. Right-click on the icon for **DataRepeater.exe** and choose Copy.
3. On your Windows Desktop, right-click (make sure you aren't right-clicking on any icons) and choose **Paste Shortcut**.
4. Right-click on the newly created shortcut and choose **Properties**.
5. On the window that appears, go to the **Shortcut** tab and click into the **Target** field and add " /SESSION=2" to the end of the path as shown in the image at right (note the space between the .exe and the /SESSION=2). Don't click **OK** yet…



6. If you want to rename your shortcut icon to reflect which session this shortcut is going to open, go to the **General** tab. At the top of this tab is a field containing the current name of the shortcut. Click into this field and re-name it as you please.
7. Click **OK** to save your changes and close out of the **Properties** window. You should now have an icon that opens a second session of Data Repeater with its own unique settings



You can create up to 9 unique session shortcut icons, numbered 1 through 9. Simply substitute the digit you desire rather than "2", as used in the example steps above.

If you open a second (or subsequent) instance of Data Repeater without using the session shortcut steps above, it'll open a duplicate of session 1, which could cause a conflict if it tries to use the same TCP ports and/or Serial ports that are already in use by the first session.

Appendix 1: Available Meta Variables

Meta variables are available to use in the parsed output templates for TCP/UDP, HTTP, HTML, Command Line, and Serial output. Below is a quick reference for all the meta variables available. They are also listed individually in the instructions in this manual for each output template type's setup section.

TCP/UDP, HTTP, Command Line, and Serial Output Meta Variables

- %ARTIST%
- %TITLE%
- %ALBUM%
- %CATEGORY%
- %LENGTH%
- %LENGTHMILS%
- %LENGTHSECONDS%
- %FILENAME%
- %URL%
- %PUBLISHER%
- %COMPOSER%
- %GENRE%
- %YEAR%
- %COMMENTS%
- %STATIONID%
- %COPY%
- %DESC%
- %PROGRAMLOG%
- %CUSTOM1%
- %CUSTOM2%
- %CUSTOM3%
- %CUSTOM4%

The HTTP Call output also has %USERNAME% and %PASSWORD% available, which use the corresponding output profile's entry from the **Username** and **Password** fields.

HTML Generator Meta Variables

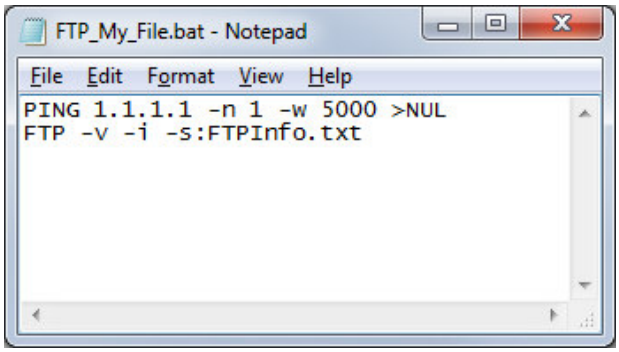
• <!--BSIARTIST-->	The artist/advertiser from the current input data
• <!--BSITITLE-->	The title/description from the current input data
• <!--BSICURRENT-->	The play time, artist, and title from the current input data
• <!--BSICURRENT1-->	The artist and title from the current input data
• <!--BSICURRENTFILE-->	The filename from the current input data
• <!--BSIALBUM-->	The album name from the current input data
• <!--BSICOPYRIGHT-->	The copyright from the current input data
• <!--BSICOMMENTS-->	The comments from the current input data
• <!--BSICOMPOSER-->	The composer from the current input data
• <!--BSIYEAR-->	The year from the current input data
• <!--BSIGENRE-->	The genre from the current input data
• <!--BSIPUBLISHER-->	The publisher from the current input data
• <!--BSIURL-->	The URL/Website from the current input data
• <!--BSIALBUMARTFILENAME-->	The album art file name from the current input data*
• <!--BSIRECENT-->	A long format list of all the recent items played
• <!--BSIRECENTX1--> through <!--BSIRECENTX##-->	Lists an individual recent item's play time, artist name title, and year (where ## is the specific recent item's number, without leading zero)
• <!--BSIRECENT1-->	A short format list of all the recent items played
• <!--BSIRECNET1X1--> through <!--BSIRECENT1X1##-->	Lists an individual recent item's artist name and title (where ## is the specific recent item's number, without leading zero)
• <!--BSIRECENTFILES-->	A list of the filenames of the recent items played
• <!--BSIRECENTFILES1--> through <!--BSIRECENTFILES##-->	Lists an individual recent item's file name (where ## is the specific item's number, without leading zero)
• <!--BSIWEATHER-->	A multi-field listing of the current weather information
• <!--BSIWEATHERAVAILABLE-->	Displays Yes/No if weather info has been retrieved
• <!--BSIWEATHERTIME-->	Time weather info was updated by wunderground.com
• <!--BSIWEATHERCITY-->	City weather info is for
• <!--BSIWEATHERSTATE-->	State weather info is for
• <!--BSIWEATHERTEMPERATURE-->	Current temperature
• <!--BSIWEATHERCONDITIONS-->	Current sky conditions (cloudy, sunny, calm, etc.)
• <!--BSIWEATHERWINDS-->	Current Wind direction and speed
• <!--BSIWEATHERPRESSURE-->	Current barometric pressure of the atmosphere
• <!--BSIWEATHERHUMIDITY-->	Current percent humidity
• <!--BSIWEATHERVISIBILITY-->	Current visibility distance and conditions
• <!--BSISTATIONID-->	Station ID text (from input data, or from default entry)
• <!--BSITIME-->	System time on the computer running Data Repeater
• <!--BSIDATE-->	System date on the computer running Data Repeater

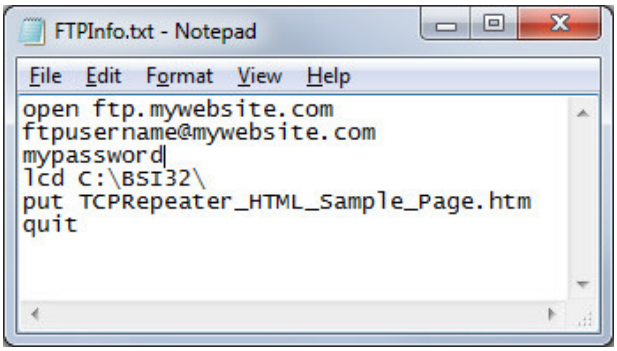
Appendix 2: Using a DOS Batch File to FTP your HTML page

In addition to the built in **Auto** FTP function of the HTML Generator, Data Repeater's Command Line function can be used to FTP your web page file created using Data Repeater's HTML Generator function. A DOS Batch file can be employed that uses Windows' built in FTP function to send the file to your web server so your website automatically has the most up-to-date version of your HTML file each time a new version is generated as data is received by Data Repeater.

Since we need the batch file to delay sending the file for a few seconds so that Data Repeater has time to finish creating the web page with the HTML Generator, we will also use a workaround with DOS's PING command that causes it to create a delay before moving on to the FTP command in the batch file.

The images below show the files required to run the process, the batch file itself, and the FTP function's connection info document. Both the batch file and the FTP info document are created using an ASCII text editor, such as notepad.exe, which is included with Windows.





The first command in the batch file, "PING 1.1.1.1 -n 1 -w 5000 >NUL", causes the PING command to try to ping an impossible IP address, and wait to time out for 5000ms (or 5 seconds) before finishing and letting the batch file continue, at which point it runs the FTP command.

The FTP Command runs using the info contained in the FTPInfo.txt file in the same folder as the batch file. Given the instructions listed in the FTPInfo.txt file shown above, the FTP function logs into the FTP site ftp.mywebsite.com using the username ftpusername@mywebsite.com and username "mypassword". It then moves it's current local directory to the C:\BSI32 folder (in this example, Data Repeater is outputting the HTML destination page, TCPRepeater_HTML_Sample_Page.htm, to the C:\BSI32 folder), and then it transfers the TCPRepeater_HTML_Sample_Page.htm from the C:\BSI32 folder to the root of the FTP directory on the web server, then the FTP function quits.

Note that the FTP function built into Windows does not handle passive FTP servers, which are becoming quite common. An open source command line FTP client that supports passive connections and binary transfers, called "ftps.exe", is set up in a very similar fashion. You can search for "ftps.exe" on Google, or your favorite search engine to find this free application and instructions on how to set it up for executing from a batch file.

Technical Support

Regular Technical Support hours are 6am to 6pm (PST / PDT) from Monday through Friday. Telephone support for Data Repeater is available with a BSI Platinum or Gold Tech Care Plan, or on a 'per incident' basis. For more information on our Tech Care Plans, check out our website at http://www.bsiusa.com.

If you have a technical question or problem and do not have a Gold or Platinum Tech Care Plan, the best way to get in touch with us is via our Support Center on-line at http://support.bsiusa.com

Limited Warranty

Broadcast Software International warrants that all disks provided are free from defects in material and workmanship, assuming normal use, for a period of 90 days from the date of purchase.

Broadcast Software International warrants that the program will perform in substantial compliance with the documentation supplied with the software product. If a significant defect in the product is found, the Purchaser may return the product for a refund within 15 days of purchase. In no event will such a refund exceed the purchase price of the product.

EXCEPT AS PROVIDED ABOVE, BROADCAST SOFTWARE INTERNATIONAL DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE PRODUCT. SHOULD THE PROGRAM PROVE DEFECTIVE, THE PURCHASER ASSUMES THE RISK OF PAYING THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION AND ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES. IN NO EVENT WILL BROADCAST SOFTWARE INTERNATIONAL BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING WITHOUT LIMITATION DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION AND THE LIKE) ARISING OUT OF THE USE OR THE INABILITY TO USE THIS PRODUCT EVEN IF BROADCAST SOFTWARE INTERNATIONAL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Use of this product for any period of time constitutes your acceptance of this agreement and subjects you to its contents.

Contacting BSI

BSI - Broadcast Software International
909 International Way
Springfield, Oregon USA 97477

Headquarters: (541) 338-8588
Headquarters Toll-Free: (888) 274-8721
Headquarters Fax: (541) 338-8656
Headquarters email: info@bsiusa.com

Sales: (541) 338-8588
Sales Fax: (541) 338-8656
Sales email: sales@bsiusa.com

Platinum & Gold Technical Support: (541) 342-5250 (for customers with a Tech Care Plan only)
Standard Technical Support Center (Support Ticket): http://support.bsiusa.com