Regression Report Tool

Release (1.12.0)

Jeffrey Wren

Regression Report Tool

Release (1.12.0)
Jeffrey Wren

Published January 30, 2009

Paradigm Works, Inc.

300 Brickstone Square
suite 104
Andover, MA 01810

Website: http://www.paradigm-works.com
Email: <pw-support@paradigm-works.com>

Table of Contents

List of Figures

List of Examples

Chapter 1. Overview

1.1. Introduction

The Regression Result Tool (RRT) is a Perl based tool that displays regression results via a web page. It takes as input a directory/list of test log files. It parses each for pass/fail information and any other information the user would like to have reported. It will generate an HTML report that contains links to each of the tests allowing for quick navigation (Figure 2.1: Regression Report Web Page (page10)). Finally, it automatically updates a master/home page where links to each of the reports resides (Figure 1.1: Main Report Web Page).

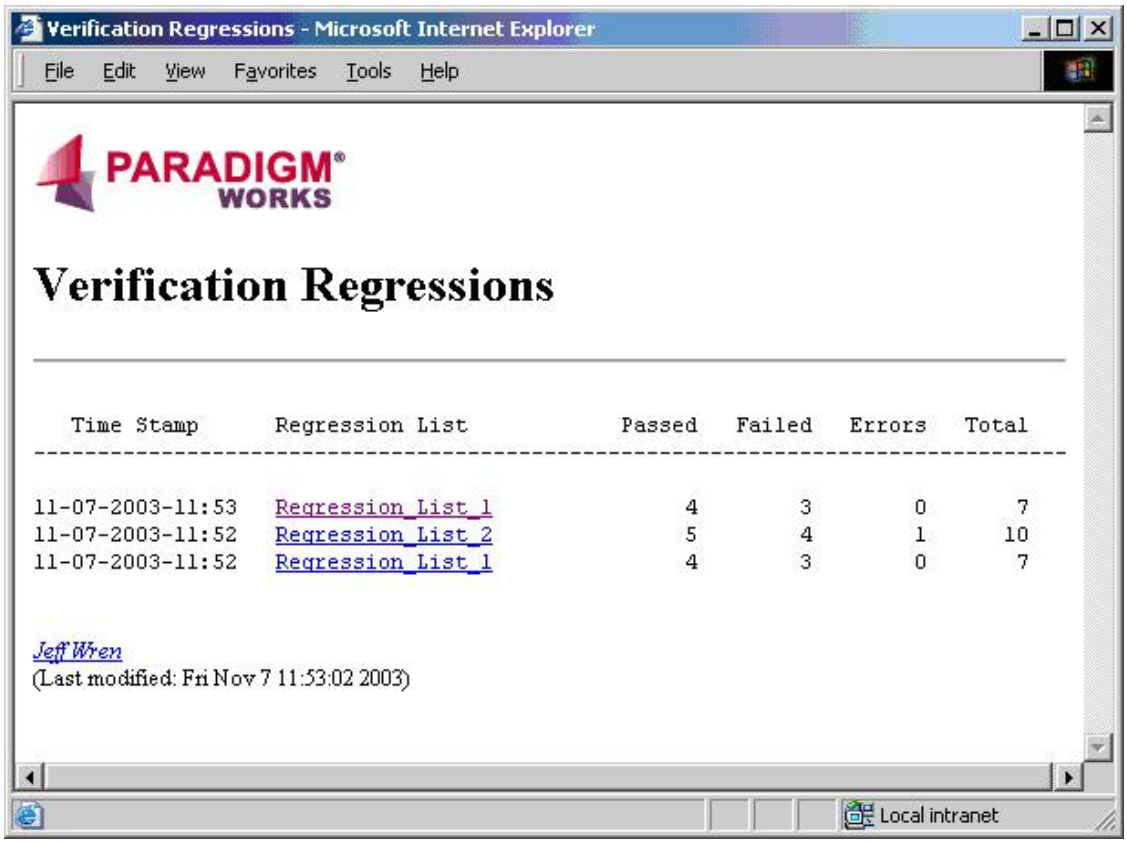

Figure 1.1. Main Report Web Page

1.1.1. Features

- Specify a directory, a list of files, list of directories, etc for parsing
- Parsing of test results and report displaying is controlled via a configuration file
- Supports multiple configuration files
- Generates HTML reports with highlighted pass/fail information

- Generates a text only report
- Web Links to individual test result log files
- Tabulation of Pass/Fail data
- Email notification support via sendmail, smtp, or qmail
- Archiving of old results and maintaining of such on separate web page
- Deletion of old results
- Options to limit size of stored log files (runaway tests)

1.1.2. Requirements

- Perl 5.6.1 or Later
- A Unix based OS. This includes Cygwin
- Web browser
- During setup, an individual with a basic knowledge of Perl regular expressions
- Tar
- Gzip
- Sendmail, qmail, or smtp server (For the Email notification feature)
- Each test log file needs to have a unique name (Only, if not using the recursive mode)
- The log files that are to be parsed must have the following which can be uniquely identified with a Perl regular expression

  1. Test Name
  2. Pass/Fail Info

Chapter 2. Setup

2.1. Quick Start

The first thing is to get the latest distribution of the RRT archive. The current released version of the RRT tool is stored as a compressed tar file on the SourceForge web site (http://sourceforge.net/projects/pw-rrt) [http://sourceforge.net/projects/pw-rrt/]. Download and copy this file to your home area or other test area and run the following commands which are based on the distribution of this documents release:

```
>> gzip -d rrt-1_12_0.tar.gz
>> tar -xvf rrt-1_12_0.tar
>> rrt-1_12_0/bin/rrt -r -n rrt-1_12_0/test_logs
```

RRT is distributed with the default of the distribution's rrt-1_12_0/public_html as the location of log file storage. The above commands will parse and display the test log files to STDOUT. In order to post the log files to the web pages one would just re-execute without the "-n" switch.

```
>> rrt-1_12_0/bin/rrt -r rrt-1_12_0/test_logs
```

Using a web browser open the file rrt-1_12_0/public_html/rrt_home.htm. One should see a single link line with the time and date the regression results were parsed. The links exist. Feel free to explore. For example:

```
>> firefox file://${PWD}/rrt-1_12_0/public_html/rrt_home.htm
```

For more options to test, please see the online help screen:

```
>> rrt-1_12_0/bin/rrt -help
```

The above steps are meant to provide one with a quick way to get RRT up and running in order to see how it works. For the formal installation instructions please see the next section.

2.2. Installation

The RRT tool is written in Perl and is distributed as an archived tarball. All one needs to do to install it is to select the desired location and execute the following:

```
>> gzip -d rrt-1_12_0.tar.gz
>> tar -xvf rrt-1_12_0.tar
```

Perform the following steps to set-up basic functionality:

1. Copy the public_html directory to the desired location where one wants the reports and test log files to be stored. One can rename this directory to something else.

2. Copy the rrt.cfg file to a new name:

   ```
   >> cp rrt-1_12_0/lib/rrt.cfg my_rrt.cfg
   ```

3. Edit my_rrt.cfg.
   - Update the "master_web_dir" parameter to point to the location one copied the public_html directory in step 1.
   - Ask an IT admin to set-up a web link to the location that one copied the public_html directory to. This is needed if one wants to use HTTP. If one is going to use file links, then one can skip this.
   - Set the parameter master_web_link to the HTTP or file link.
4. Add the rrt-1_12_0/bin location to ones path
5. Execute the following to test your set-up:

   ```
   >> rrt -config my_rrt.cfg -rec -no <dir_to_parse>
   ```

6. Verify that the test information is what one desires. If it is not, then modify my_rrt.cfg and re-execute step 5 until the needed results are achieved.
7. Once the desired information is captured, one can execute the following to store the information into the public_html area that was set-up in step 1.

   ```
   >> rrt -config my_rrt.cfg -rec <dir_to_parse>
   ```

2.3. Environment Variables

RRT_HOME	This is a special environment variable that is set at execution time by the **rrt** tool. It is the full path to the distribution directory that the **rrt** command was executed from. For example. If the path to the bin directory of rrt is /home/user/rrt/bin, then RRT_HOME would be set to /home/user/rrt.
RRT_CONFIG	This environment variable can be set to the default RRT configuration file to use. If not set the default configuration file that is parsed is rrt/lib/rrt.cfg. The command line option `-config_file` overrides this environment variable setting.
RRT_PERL	The path to the version of Perl that one would like to use. RRT requires Perl version 5.6.1 or greater. The format of the parameter is just the path, not the Perl executable. For example, given that Perl lives at /usr/local/bin/perl_5_6_1/perl, the environment variable would be set to /usr/local/bin/perl_5_6_1/ (The ending "/" is required).

2.4. Configuring the rrt.cfg File

The configuration file is used to specify one's unique environment. One can setup email addresses, report formatting, search parameters, and environment info. Provided with the RRT tool is an example configuration file. It resides in the rrt/lib directory. One can copy this file for modification, or update the one provided directly. If a copy is made, be sure to use the `-config` file option when calling the tool. The following sections detail the options that can be controlled.

2.4.1. Email Notification

The following is a list of all the parameters available for configuring email notifications:

email_program — The mail program to use for email notifications. Valid entries are:

- sendmail
- qmail
- smtp

If **sendmail** or **qmail** is desired, set the parameter *email_program* to just the name of the tool, e.g. **sendmail**. If smtp is desired then set the variable *email_program* to smtp. Also, for smtp to work the parameter *email_smtp_server* must be set to the domain name that the server lives on. The script **test_email** is provided to test the different options in order to find one that works. It has a help splash screen available with the -help option.

email_smtp_server — Domain where the smtp server lives on, e.g. mail.usgov.gov. This must be set if the selected *email_program* is smtp. This parameter is ignored if anything other than smtp is set.

email_domain_name — This parameter is provide as a way to append a domain name to addresses that do not have one. This typically occurs when one is using a username as the address to send to. The domain will only be appended to addresses that do not already have one.

email_addresses — When using the email notification feature, one can specify who will be sent a message by maintaining a list of addresses in the configuration file. The keyword *email_addresses* is used to designate that a list of addresses is to follow. Each entry should be separated by a ",". The last entry in the list should end with a ";". Spaces and line feeds are ignored. There should be only one instance of *email_addresses*. If there is more than one, the last one will be the list of addresses used.

Example 2.1. Specifying Email Addresses in Configuration File

```
email_addresses George.Washington@usgov.gov, Abe.Lincoln@usgov.gov,

John.Kennedy@usgov.gov, RichardNixon@usgov.gov;
```

2.4.2. Log File Extensions

A configuration file parameter is provided that allows one to specify what file extensions are being used to designate a test log file. This is a comma separated list where all the items must be contained on a single line. Note, extensions are not limited to just file types. One can specify a full and partial file names as well.

Example 2.2. Specifying Log File Extensions to Search and Parse

log_file_extensions .log, .txt, sylver.log, _compile.log

2.4.2.1. Support for files that end ".gz" (gzip)

As of version 1.11.0 rrt is capable of parsing log files that have been compressed using the gzip tool. This is to support those verification environments that automatically compress their regression log files. Rrt will properly parse files that end in ".gz" and will store them in the rrt_logs directory. The files are stored uncompressed because web servers do not always automatically extract the files properly for viewing. When using the rrt_max_file_size parameter, it should be noted that the size of the file when it is uncompressed is the value that is used to determine if the file is too large.

2.4.3. RRT Environment Configuration Parameters

The following parameters in the configuration file are used to designate such things as where the web pages, test logs, and reports are stored:

master_web_dir	This is the top level directory where the web pages, reports, and test logs will be stored.
master_web_page	Name of the web home page file. This can be any web page. A template web page is provided (rrt_home.htm). The key is to have the following three lines placed in the HTML file. The results will be placed between these lines. <!-- Regression Results Header --><PRE> <!-- Regression Results Start --> <!-- Regression Results End --></PRE>
archive_web_page	Name of the archive web page. This is the page that regression results are moved to before deletion. The individual report pages are available. However, the links to the each of the log files is disabled. A template web page is provided (rrt_archived.htm).
master_web_link	This is the link string that one would enter into the browser to access the web pages minus the *master_web_page* name.
log_archive_path	Location where test log files will be stored. This is a path that will be placed under the *master_web_dir* location. Therefore, if the storage area needs to be located elsewhere, then use a symbolic link to the desired destination.
report_archive_path	Location where report files will be stored. The same restriction applies to this directory as specified in the log_archive_path parameter.

There are two methods that one can use to store and access the web pages. One is through a web server. The other is to directly point to the web home page file directly. In the first method every individual who wishes to access the information does not have to know or have access to the file system, just the web server. In direct file mode, each person who wants to access the results must be able to read from the directory location that the files are maintained.

If one is working on a system that has an Apache web server one can just copy the directory "rrt/public_html" to "~/". Then, the web pages can be accessed via the link `http://<host>/~<user>/<rrt_home_page>`. If one is using a different web server, does not have a user account on the web server machine, or just does not wish to use their personal home directory, then one will have to talk to the web administrator to set up a link.

2.4.4. Performance Configuration Parameters

Rrt provides several parameters to reduce the time it takes to search for, parse, and store log files. The following can be defined in the configuration file.

rrt_fast_mode — Parameter that allows one to set the parsing mode that RRT uses when parsing a file. The default is to parse the entire log file. However, the more search keys one has the longer it takes for RRT to parse the file. Since in 95% (or higher) of the time the information that one is parsing for is at the beginning and end of the log file, parsing the middle of large files is needless. Therefore, the following fast mode allows one to tell RRT to only look at the top x bytes of the file as well as only the last x bytes of the file. If you need to parse the entire file then comment out the line, or set the value to 0.

rrt_max_file_size — Occasionally, in a verification environment a test will get into a mode where the simulation only ends after a watchdog timeout. This can result in an output log file that exceeds 1 Gigabyte or more. These files can be time consuming to parse and they can also use up much needed disk space because rrt will store them for access via the web interface. The rrt_max_file_size parameter will set a maximum limit that rrt will use to parse and store a file. If a file exceeds this limit, rrt will only parse and store the upper rrt_max_file_size / 2 bytes of the file. It will also parse and store the lower rrt_max_file_size / 2 portion of the file. The middle portion of the file will not be parsed and will be discarded. The parameter is specified in number of Megabytes. The default is 20 MBytes

Note

If the rrt_fast_mode parameter is defined then it determines the amount of the file that is parsed regardless if the file is too large. However, the storage rule will still apply.

include_dir_types — This is a comma separated list of directory names to include in the search for log files. The directory names do not need to be complete. The search will match partial names. For example, if all log files are stored in run_<seed> directories then set the include value to be run_. The result will be only those paths that have “run_” in their name will be searched. This parameter only applies when using the rrt –recursive command line option.

```
include_dir_types run_
```

exclude_dir_types — This parameter allows one to define a comma separated list of directory names to exclude from the search for log files. If the name appears anywhere in the path it is excluded. Note, Exclusion has precedence over inclusion using the include_dir_types parameter. This parameter only applies when using the rrt -recursive command line option.

```
exclude_dir_types work, src
```

rrt_num_not_to_archive — The **rrt** `-archive` and `-delete_archive` option commands are based on the date the regressions were parsed. As a project ma-

tures, regressions may stop or only be run occasionally. If a cron process has been setup and it has not been disabled the result would be that when someone comes back to review the regression results, everything is gone because the date thresholds were met.

The *rrt_num_not_to_archive* parameter allows one to specify a minimum number of regressions to be kept on the master web page regardless of their date-stamp. These reports will be maintained on the main page and all of the log files will still be accessible. Default: 10

rrt_reports_to_keep — This parameter specifies the number of reports to maintain on the archive web page regardless of their date-stamp. Only the reports themselves will be maintained. Any archived log files will still be removed. Default: 20

2.4.5. Search Strings and Web Report Fields

Special parameters are provided that can be used to format the output information that is displayed on each of the web pages. The following parameters are used in the configuration file to define the action that is to be performed on the variable that comes after.

search_key — Assigns to a defined or user specified variable the pearl search string to use to capture data. This is the method in which way one details what information one would like to retrieve from each of the test result files. The variables defined here are then used to populate the master & report fields. See Section 2.4.6: Predefined Variables (page10) for a list of predefined variable names.

master_field — Designates the title, content, width, and justification of a column field on the master/home web page. See the below report field example for the format.

report_field — Designates the title, content, width, and justification of a column field in the report files.

Example 2.3. Setting Up Search Keys

The following shows the assigning of Perl regular expressions to variable names. Note how these variables are used in the next example to set up the report fields.

```
search_key    test_name_string   "/testfn=(\w+)/"
search_key    passed_string      "/(PASSED)/"
search_key    failed_string      "/(FAILED)/"
```

Note

The above variables are **rrt** defined and are the minimum set that need search definitions in order for the tool to work properly. Any other search key variables are considered user defined and can be anything the end user wishes. Also, if the “test_name_string” is not matched in the log file then the log file is considered invalid and is removed from the report. This is to account for those environments that may have .log files that are generated by different tools (See Sec-

tion 2.4.6: Predefined Variables (page10) for a list of predefined variables).

Example 2.4. Main Web Page Report Set-Up

Setting up the master fields. This is an example of how to set up the configuration file to display tabulated regression results data on the main web page. See Figure 1.1, “Main Report Web Page” [1] for the subsequent output.

Field Type	Keyword	Column Header	Col	Width	Justification (L/R/C)
master_field	date_stamp	"Time Stamp"	1	16	C
master_field	list_name	"Regression List"	2	24	L
master_field	pass_total	"Passed"	3	6	R
master_field	fail_total	"Failed"	4	6	R
master_field	error_total	"Errors"	5	6	R
master_field	test_total	"Total"	6	5	R

Example 2.5. Test Report Set-Up

Setting up the report fields. This is an example of how to set up the information that will be displayed on the individual test results page. See Figure 2.1: Regression Report Web Page (page10) for the resultant output.

Field Type	Keyword	Col	Column Header	Min Width	Justification (L/R/C)
report_field	test_name_string	1	"Test name"	16	L
report_field	user_defined_1	2	"Configuration"	16	L
report_field	passed_string	3	"Result"	6	L
report_field	failed_string	3	"Result"	6	L
report_field	error_string	3	"Result"	6	L
report_field	test_duration	4	"Duration"	8	R

Note

In the above example, the `passed_string`, `failed_string`, and `error_string` variables share a common column number. This overloading works because only one should be true for any given test.

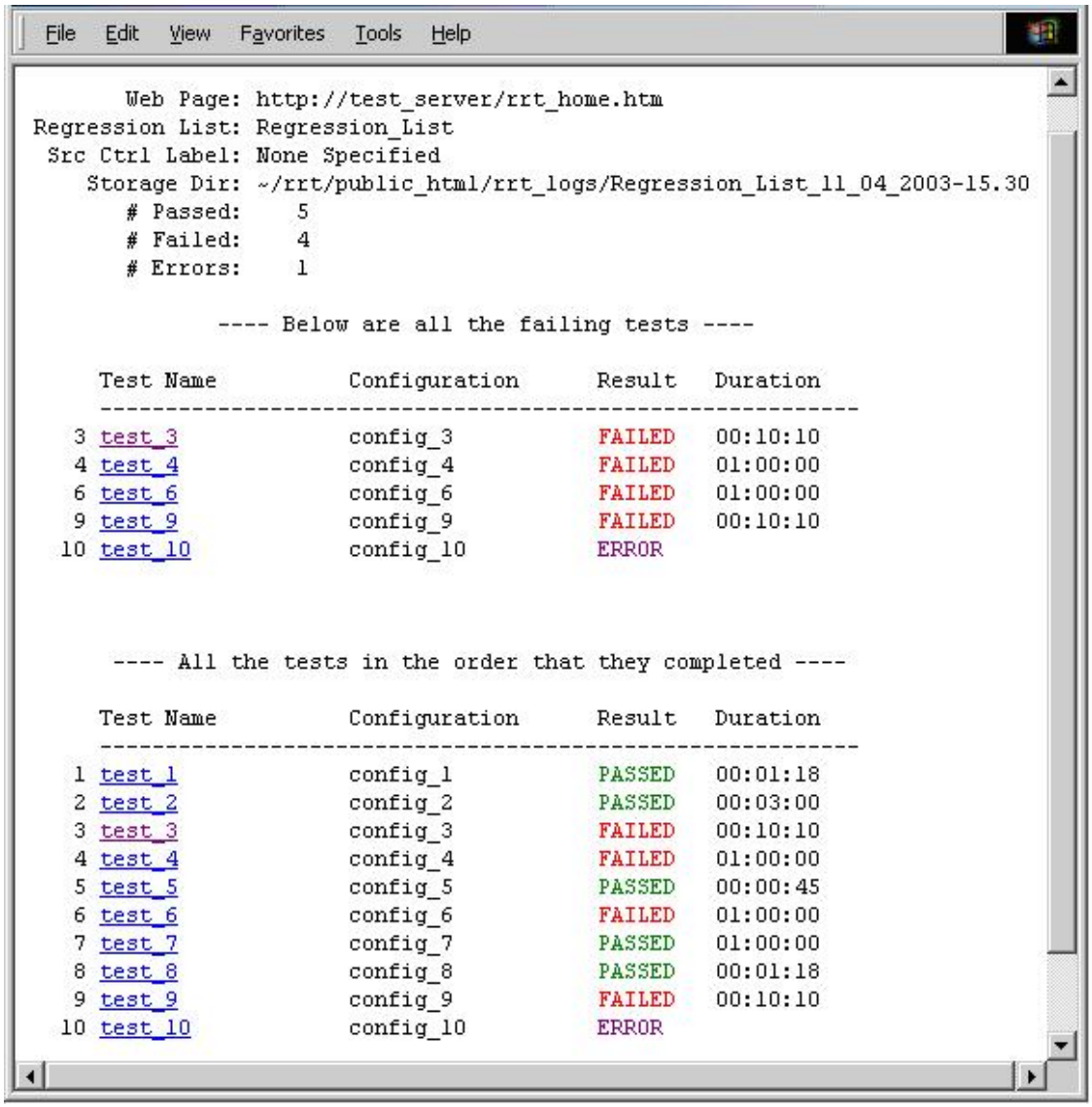

Figure 2.1. Regression Report Web Page

2.4.6. Predefined Variables

RRT has a set of variables that are available for use in the output reports. The `test_name`, `passed_string`, and `failed_string` must be set in the configuration file by the user for the tool to work properly. The remaining variables are controlled by RRT and contain information relevant to the regression.

test_name (Required)

This variable is set for each log file that has a Perl regular expression match using the “search_key” mechanism. It is provided as a means of capturing the name of the test that results are being retrieved for. If no test name is matched then the log file is removed from the report. This is to account for those environments that

	may have .log files that are generated by different tools. For each match the test_total counter is incremented. This variable is typically used in the report_fields reports (Figure 2.1: Regression Report Web Page (page 10)). It is suggested that the test name be located at the start of the log file. That way if the test errors out due to compile or system issues, RRT will not delete the log file because it could not find a test name
passed_string (Required)	This variable is set for each log file that has a Perl regular expression match using the "search_key" mechanism. It is provided as a way of determining whether a test ran to completion and "PASSED". For each test that passes, the pass_total counter is incremented. This variable is typically used in the report_fields reports (Figure 2.1: Regression Report Web Page (page 10)).
failed_string (Required)	This variable is set for each log file that has a Perl regular expression match using the "search_key" mechanism. It is provided as a way of determining whether a test ran to completion and "FAILED". For each test failure the fail_total counter is incremented. This variable is typically used in the report_fields reports (Figure 2.1: Regression Report Web Page (page 10)).
error_string	This string has the constant value of "ERROR". It cannot be set via the search_key mechanism. It represents the failure case where a test_name string was found but no pass/fail string was matched. This provides a means of categorizing tests that did not run to completion, e.g. Compilation errors, segmentation faults, etc. Each time no pass/fail info is found in a log file the error_total counter is incremented. This variable is typically used in the report_fields reports (Figure 2.1: Regression Report Web Page (page 10)).
date_stamp	If not set from the command line this variable represents the time that the RRT tool was called to perform its task. The default format is MM_DD_YYYY-hh.mm. It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).
list_name	Represents the name of the list of tests that were run. It is set from the command line. The default is "Regression_list". It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).
pass_total	This is a counter that represents each of the passed_string matches, i.e. The tests that passed. It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).
fail_total	This is a counter that represents each of the fail_string matches, i.e. The tests that failed. It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).
error_total	This is a counter that represents the number of log files that did not have a pass or fail match. It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).
test_total	This is a counter that represents each of the test_name matches, i.e. The number of tests that were run. It is typically used on the master report page (Figure 1.1: Main Report Web Page (page 1)).

Chapter 3. Command Line Options

3.1. Address Override

This option is provided as a means to override the email addresses that are set within the configuration file and to send an email notification to one or more individuals. A typical application is where a common configuration file is used for parsing data; however, the parsing of test results is for a particular individual and not the entire team. Use multiple `-address` options to specify multiple people.

Note

The `-address` option only works in conjunction with the `-email` option.

```
>> rrt -address user_1 -addr user_2 regression_dir
```

3.2. Archiving

As time goes by the master web page can become cluttered with many lines that contain links to information that is of little interest. However, it is not desirable to remove the information completely. An archive option is provided that will tar & compress the report log files, remove the links from the report files to the individual test logs, and move the report links from the home page to an archive page (See Figure 3.1: The Archive Web Page (page13)). The archive option takes a single argument of the number of weeks to keep. That is anything older than the number of weeks from the current date will be archived. In the first example anything older than 2 weeks will be archived. In the second, anything older than a ½ week will be archived.

```
>> rrt -archive 2
>> rrt -archive 0.5
```

3.3. Configuration Files

The configuration file switch (`-config_file`) is provided to allow one to maintain multiple config files. This provides one with the option of maintaining multiple report types, web pages, email lists, etc. The default is to use the configuration file lib/rrt.cfg.

```
>> rrt -config_file my_config.cfg regression_dir
```

3.4. Date Stamps

The `-date_stamp` option is used for two purposes. One, as a point of reference to the time the regression results were generated. Two, as a means to uniquely store results that have the same regression list name (See Figure 1.1: Main Report Web Page (page 1)). The RRT default is to generate a date stamp of the form MM_DD_YYYY-hh.mm at the time the tool is called. However, if one would like to use a different unique labeling system one can override this behavior by supplying one's own date stamp. Note: since the date stamp is used in creating files and directories it must contain only those characters that are acceptable for such.

```
> rrt -date_stamp 2002-21-02 regression_dir
> rrt -date_stamp Mon_Feb_21_02 regression_dir
```

Figure 3.1. The Archive Web Page

3.5. Deleting Old Archives

After a certain period of time, the archived results that are being maintained are no longer needed. A delete_archive option has been provided that will remove the archived test logs (tar.gz). The associated report web pages will be maintained. In order to remove the reports from the web pages use the delete_reports option in conjunction with the `-delete_archives` option. Just like in the archive option this command takes as an argument the number of weeks to keep.

```
>> rrt -delete_archives 4

>> rrt -da 12
```

In the first example any archived folder (tar.gz) older than 4 weeks will be deleted. In the second, anything older than 12 weeks will be deleted.

3.6. Deleting Old Reports

After a certain period of time, the report results that are being maintained on the archived web page are no longer needed. A `-delete_reports` option has been provided that will remove the report files and the report links from the effected web pages. This option must be used with the `-delete_archives` option. In the following first example, anything older than 12 weeks is deleted and removed from the web pages and log storage area.

```
>> rrt –delete_archives 12 –delete_reports
```

In the next example, the command is executed twice, once where the archived log files are removed after 2 weeks. The second call removes the report files after 1 year. This approach is advantageous where one would like to keep a projects regression history over a long period of time without keeping all of the compressed log files.

```
>> rrt –da 2
>> rrt –da 52 -dr
```

3.7. Email

One of the main purposes of the RRT web tool is to allow an entire group to have quick access to the regression results. The Email feature is provided to notify the appropriate individuals when there is new data for review. The address list is maintained in the configuration file or by specifying the `-address` <email_address> option on the command line. In order for the email to be sent, one must specify the `-email` option on the command line.

```
>> rrt -recursive -email rrt/test_logs
```

3.8. Force

Perform the deletion of archived results without a confirmation prompt. This is provided for the scenario when one has automated mechanism for deleting old archives, such as with cron.

3.9. List Name

An option is provided to allow one to label the regression results with a user defined name. This is typically the name of the regression list. The default is to label the results “Regression_list” (See Figure 1.1: Main Report Web Page (page 1)).

```
> rrt -list_name Switch_List regression_dir

> rrt -list_name “Memory Controller List” regression_dir
```

3.10. No Web Page

The `-no_web_page` option disables the update/creation of the report web pages and prints the regression report to STDOUT.

3.11. Parse Only

Parse the test results and return pass/fail data only. Do not store the data, update the web pages, or send email notifications. The output is displayed to STDOUT. The example below is using the example log

files that are included in the RRT distribution.

```
>> rrt -parse_only rrt/test_logs

Total tests that passed  = 4
Total tests that failed  = 3
Total tests that errored = 0
Total number of tests    = 7
```

3.12. Recursive

To parse the input directory recursively use the `-recursive` switch. This allows for a complex test environment and for test names to have the same file name. The uniqueness is in the path name. The first command line argument must be a directory. Any other arguments following this will be ignored. The example below is using the example log files that are included in the RRT distribution.

```
>> rrt -parse_only -recursive rrt/test_logs

Total tests that passed  = 5
Total tests that failed  = 4
Total tests that errored = 1
Total number of tests    = 10
```

3.13. Source Control Labels

A switch has been provided to enable one a method to specify what the source control label/tag that is needed to reproduce the regression environment. This string is output on the report page on the output line “Src Ctrl Label” (See Figure 2.1: Regression Report Web Page (page 10)). The default is “None Specified”.

```
>> rrt -src_ctrl_label Release_1_0 regression_dir
```

Chapter 4. Support Scripts

RRT is distributed with a couple of support scripts to help with setup and test log file capture.

4.1. rrt_test_wrapper

The script **rrt_test_wrapper** is a utility script to provide RRT needed header and PASS/FAIL info so that RRT can parse the log files for test results. One, would use this script in situations where a test script does everything that is needed, but is missing the required RRT fields. This script can also be set up to monitor for error strings and fail the test appropriately. For Example:

```
>> rrt_test_wrapper -test_name my_test \
                    -define "Configuration = testbench_a" \
                    -log sim/my_test.log \
                    command_to_execute -command_options
```

For a full list of options execute the command with the `-help` option.

4.2. test_email

This script is provided in order to allow one to test the different email options to determine the correct tool and configuration to use. Based upon the options that one sets on the command line, a sample email will be sent. One can then verify that the email was actually received. Once the email is working, one can then take the options used and update the rrt.cfg file with the appropriate configuration information.

For a full list of options execute the command with the `-help` option.

Appendix A. Release Notes

- Release 1.12.0
  - Added new RRT_HOME env variable that is set at execution time to the directory that RRT was executed from.
  - Added ability to expand environment variables in the RRT configuration file (rrt.cfg).
  - Added in the Perl Mail package in order to expand RRT email capabilities to include qmail and smtp. Previously, only sendmail was supported.
  - First official SourceForge Release.
- Release 1.11.1
  - Fixed issue where the log_file_extensions could match anywhere in the filename. Now, they match from the end of the filename only
  - Fixed issue where the first line of the first file loaded was not being parsed. This could cause issues if the log file had the testname on the first line. This problem was introduced in release 1.11.0 with the parsing performance enhancement.
  - Under the right conditions, the column overloading used in the regression report for the PASSED/FAILED/ERROR strings could result in multiple result columns being displayed in the same report. This has been fixed.
- Release 1.11.0
  - Added `-delete_reports` option functionality. Now the web reports can be kept even if the archived tar.gz files are to be removed. Previously, the `-delete_archives` option removed both.
  - Changed the way the date of the files is determined. Previously, the system time-stamp of the file was used. This was not robust enough because if a user copied or moved the location of the files, the dates could be messed up resulting in non-deterministic behavior. Now, the date-stamp in the filename is used to determine if the file should be archived/removed.
  - Modified the way parsing of the log files is performed. This should be a performance enhancement. Preliminary testing shows at least a 30% speed boost.
  - New rrt_max_file_size parameter has been added that limits the maximum size that a file can be for parsing and storage. If a file exceeds this size, then RRT will only parse and store the upper and lower rrt_max_file_size/2 portions of the file.
  - Added support for log files that have been compressed using the tool gzip. These files will be parsed using the **gzip -dc** command. They will be stored uncompressed for web access.
  - Added two new parameters *`rrt_num_to_not_archive`* and *`rrt_reports_to_keep`*. These allow one to set a minimum number of reports to be maintained on the main web page and the archive web page regardless of their date-stamp. This will prevent a cron process or user from removing all of the history information for a project even if a project becomes idle and the date-stamp thresholds are met.
  - Updated documentation

- Release 1.10.1
  - Updated the documentation and release notes.
- Release 1.10.0
  - Added the Apache 2 license. The Regression Report Tool is now available as open source.
  - Previously the passed/fail string that was displayed in the user report was whatever was captured via the regular expression. Now, the fixed strings of PASSED/FAILED will be used regardless of what the user's pass/fail string is in the log file.

Appendix B. GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Ver-
. sion.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that

is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (C) YEAR YOUR NAME.
>
> Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.