ZebraTester

Release Notes

Version 5.5-A

Table of Contents

All Rights Reserved

1 V5.5

1.1 New Functions and Features

1.1.1 Major Release V5.5-A (Installation Kit dated February 1, 2017)

ZebraTester V5.5-A contains the following new features:

- **Enhanced Session Cutter**
  The "Session Cutter" menu that allows you to compile fragments of different recorded sessions into a new session has enhanced in such a way that now existing definitions of Variables, Variable Extractors and Variable Assigners are copied into the new compiled session. In addition, the name of the Page Breaks can now modified in this menu and a preview of all Variables, Variable Extractors and Variable Assigners is shown per loaded session.

- **Support of WebSockets**
  The recording and executing of Web Sessions containing WebSockets messages as described in RFC 6455 is now supported. In order to avoid endless recording of WebSocket streams the recoding time can configured in the "Personal Settings" menu. The recorded WebSocket frames are shown in the "URL Detail" menu where you can also define Var Extractors and Var Assigners for the WebSocket frame data. The WebSocket frames transmitted during the load test can verified either by counting the number of frames of by searching for text patterns that should occur inside the frame data.

- **Import of LoadRunner Scripts**
  LoadRunner Scripts (written in C language) can now converted automatically into ZebraTester sessions. The load LoadRunner Scripts can placed into any "Project Navigator" directory from which they can converted by a single mouse click. The conversion includes also correlations and transactions and can applied for LoadRunner Scripts created with LoadRunner version 11.x and 12.x. The import can also automated by calling the command line tool LRToProxyDataDump (see revised "ZebraTester Application Reference Manual").

- **3 new Types of Variable Extractors**
  The following 3 new types of Variable Extractors have been implemented: Left and Right Boundary based Var Extractors, RegEx based Var Extractors, XPath based Var Extractors. These 3 types of Extractors also support Random Extraction.

- **Support of HTTPS for the WebAdmin GUI**
  By using the WebAdmin startup argument `-https <full DNS name or IP address>` an additionally Web Admin HTTPS server is started that runs usually at TCP/IP port 7991 and supports encrypted connections. The "full DNS name or IP address" is used to generate the HTTPS server certificate on the fly – in the same way as the server certificates for the Proxy Recorder are generated.

- **Support of Multiple JobControllers**
  In order to speed up the data replication of load test jobs in geographically far locations ZebraTester supports now multiple JobControllers. Additional JobControllers can added and managed in the "Exec Agent Network Configuration" menu. This menu supports also to "ping" a JobController, to get the log-output of a JobController, and to reassign a cluster of load generators from one JobController to another one. Each JobController can serve multiple clusters. The communication protocol for a JobController can be either "plain TCP/IP" or "TLS over TCP/IP". Further information is available in the revised "ZebraTester Application Reference Manual", chapter 2.1 "Support of Multiple Cluster Job Controllers".

- **Full Encryption for Load Test Jobs**
  JobControllers and ExecAgents that are operated in untrusted environments (as for example in cloud data centers) can now started in "Encrypted Mode" which effects that all load test job files and all load test job data are stored as encrypted files only. Furthermore this option effects that also the data communication from the WebAdmin GUI to the JobControllers and ExecAgents is always encrypted, even when the "plain TPC/IP" communication protocol is used (File + Transport Encryption = Full Encryption). All cryptographic keys needed for encryption and decryption are dynamically generated and kept in memory only. Further information is available in the following documents:

  ➢ "Running ZebraTester Load Generators in Encrypted Mode" (New Document: ZebraTesterEncryptedLoadGenerators.pdf)

  ➢ In the revised "ZebraTester Application Reference Manual", chapter 2.2 "Running Full Encrypted Jobs in Untrusted Environments".

  ➢ In the revised "Load Test Plug-In Developer Handbook", chapter 3.4.3.10 "Support of Load Generators which are Running in Encrypted Mode"

  Starting from this product version, all of the ZebraTester Amazon EC2 Load Generators and EC2 Cluster Job Controllers (EC2 AMIs) are starting up in "Full Encrypted Mode" – but do not confuse this with the EC2 AMI encryption option provided by Amazon. ZebraTester encrypts the load test job files at application level, independently of the environment in which the ZebraTester product is operated. Therefore, you can use this new feature on any machine, also on such machines that are physically owned and operated by your company.

- **Copy Jobs on JobControllers and ExecAgents**
  Old Load Test Jobs can now copied directly on JobControllers and on ExecAgents into new Load Test Jobs, regardless of the current state of the old load test job ("configured", "running" or "completed"). Such new jobs are always in the state "configured" and can started in a normal way. The copy of the job data is performed remotely inside the Exec Agent (for Exec Agent jobs) or remotely inside the Job Controller (for cluster jobs).

- **New PrxJob Commands**
  The following commands have been added to the PrxJob utility:
    o copyJob
    o copyClusterJob,
    o getControllerList,
    o pingController and
    o getControllerClusterList.

  Detailed information about these new PrxJob commands is available in the revised "ZebraTester Application Reference Manual".

- **Support of Pure IP V6 Connections**
  Load Test Jobs can now executed in such a way that IP V6 is used only, without using IP V4 as a fallback. This allows you to examine the stability of IP V6 servers without any side effects. Further information is available in the online Help of the "Execute Load Test" → "Additional Options" menu. Option: `-enableIPv6 [<network-interface-name>].`

- **Configurable TCP/IP Connect Timeout**
  The TCP/IP connect timeout can now configured separately, in addition to the overall URL timeout. Further information is available in the online Help of the "Execute Load Test" → "Additional Options" menu. Option: `-tconnect <seconds>`.

- **Amazon EC2: Support of Data Centers "Ohio" and "Mumbai"**
  Predefined public AMIs for ExecAgents and JobControllers are now available in all EC2 data centers, inclusive the two new EC2 data centers "Ohio" and "Mumbai".

- **Running Job Controllers on Amazon EC2**
  The revised EC2 GUI allows you to launch either new Exec Agents only, or to launch new Job Controllers only, or to launch a new Job Controller together with new associated Exec Agents.
  Along with this provision, we also have introduced the extended GUI for launching a Job Controller and multiple ExecAgents on Amazon EC2 in just one step – by automatically combining all of them to a new load releasing cluster.

- **New XPath and Regex Functions for Inline Script**
  Two new function have added for Inline Scripts, for extracting values from JSON and XML data. These methods use XPath as argument. There is also a new function available to extract Regex groups. Further information is available in the revised "PRX Basic Language Reference Manual".

- **Support of Variables in Occurrence for 3 New Var Extractors**
  The occurrence position of the 3 new var extractors (Left and Right Boundary, Regex and XPath) can configured by selecting an already extracted/created variable.

In addition, ZebraTester V5.5-A contains the following bug fixes / improvements:

- **Fixed SSL/TLS Handshake Problem with some of the Web servers (mostly IIS).**
- **Increased maximum number of in parallel-executed URLs per Web page from 300 to 600 (for generated load test programs).**
- **Proxy Recorder bug fix for "No Next Proxy" in combination with the ZebraTester DNS stack.**
- **Improved performance for writing files in Inline Scripts.**
- **Improved External Resources Menu.**
- **Support of IP V6 DNS queries in Web Tools Menu.**
- **A new Load Test Plug-In named "Massive Peak Load" is available, delivered by the installation kit.**
- **Documentation for executing Slowloris DoS Tests is available on request.**

1.2   Upgrade Instructions

<u>Before upgrading</u> ZebraTester please <u>uninstall first your old version</u> by calling "Uninstall ZebraTester" which is located in the ZebraTester installation directory. All of your data in the /MyTests folder and your certificates are not deleted and survive the uninstall procedure.

Then install the latest ZebraTester version at the same location where the old version was installed before.

Note that a manual update of a system containing an older ZebraTester installation kit, performed by replacing only the *.jar files, normally result in a <u>damaged installation</u>!

We strongly recommend that you use always the uninstall and install procedure of the installation kit for upgrading ZebraTester.

1.2.1   Upgrade Manually Installed Systems

ZebraTester V5.5-A contains updated encryption libraries.
In case you have to upgrade a system that was manually installed, without using a ZebraTester installation kit, you have to replace the following files:

- **prxsniff.jar**
- **iaikPkcs11Provider.jar**
- **iaikPkcs11Wrapper.jar**
- **iaik_eccelerate.jar**
- **iaik_eccelerate_ssl.jar**
- **iaik_jce_full.jar**
- **iaik_ssl.jar**

These files can be taken from any installed 5.5 ZebraTester version, regardless which operating system is used.

The files **iaik_eccelerate_ssl.jar** and **iaikPkcs11Wrapper.jar** are **no longer merged into prxsniff.jar**. Please ensure that the Java CLASSPATH of your startup script contains all of the files listed above (+ the path to the "ZebraTester installation directory").

Note that you have also to update the file **javaSetup.dat** located in the ZebraTester installation directory. On all machines that run any component of ZebraTester (ProxySniffer, WebAdmin, JobController and/or ExecAgent). Add the file paths of **iaik_eccelerate_ssl.jar** and of **iaikPkcs11Wrapper.jar** to the **javaSetup.dat** settings of **javacClasspath=** and of **javaClasspath=**.

1.3 Version 5.5 Compatibility

Previous Version Data Files / Version 4.6-Z – Version 5.4-R	Compatibility with Version 5.5	
Recorded Web Surfing Sessions (*.prxdat files)	Compatible	[1] Load Generators (ExecAgents) which are **operated in Encrypted Mode** require load test programs that are generated by version 5.5. Workaround: You can load any old web surfing session (*.prxdat file) into V5.5 by using the Project Navigator. After that you have to generate and compile the load test program again before you can run it on V5.5.
Generated Load Test Programs (*.java and *.class files)	Compatible / NOT Compatible [1]	
Load Test Results (*.prxres files)	Compatible	

Previous Version Data Files / Version 4.0 – Version 4.6-Y	Compatibility with Version 5.5	
Recorded Web Surfing Sessions (*.prxdat files)	Compatible	[1] Workaround: You can load any old web surfing session (*.prxdat file) into V5.5 by using the Project Navigator. After that you have to generate and compile the load test program again before you can run it on V5.5.
Generated Load Test Programs (*.java and *.class files)	NOT Compatible [1]	
Load Test Results (*.prxres files)	Compatible	

2 V5.4

The brand of the product has been renamed from ProxySniffer to ZebraTester.

2.1 New Functions and Features

2.1.1 Major Release V5.4-A (Installation Kit dated July 22, 2015)

ZebraTester V5.4-A contains the following new features:

- **Revised Content Test Menu**
  The "Content Test Menu" is now an overlay instead of a pop-up window and has been revised and improved. It's now also supported to verify text fragments of a received HTTP response header.

- **Other Revised Menus**
  The following menus are now overlays instead of pop-up windows:

  - Cumulated Response Time per Media Type
  - Recorded URL Performance Data
  - Delete URLs
  - Delete Pages

  The filter for the hosts in the main menu is now also implemented as an overlay which contains checkboxes per host name, rather than a simple text input field as before. This new overly can activated only if the recorded session contains more than one host name.

- **Enhanced Var Extractor Wizard**
  A new variable extractor type has been implemented to extract values from unstructured HTTP responses. This new variable extractor type is no longer text-line based and uses as algorithm a search text and a relative token number, counted from the search text (in comparison to the already existing variable extractor whose token number always starts at the beginning of a line). Both type of variable extractors are now supported by the Var Extractor Wizard.

- **Reduced Size of Log Output for Load Test Jobs**
  Load test jobs are writing now only a small set of overall values about the measured data to the log, but no longer detailed error data in case when an URL call fails. This reduces the size of the job *.out file normally by more than 95%. All detailed error data are still available in the "Error Snapshots" as before. To re-enable the old functionality for the log output you can select the new debug option "debug failed loops" when starting a load test. Furthermore, it is now also possible to suppress completely to write any data to the job *.out file by using the new "Additional Option" -nostdoutlog.

- **Real-Time Control of Load Test Jobs**
  The "Abort Job" button shown during load test execution has been replaced by an overlay which allows you:
  - To abort the load test job (as before).
  - To suspend and to resume the load test job (new).
  - To increase or decrease the planned test duration (new).
  - To increase and to decrease the number of simulated users (new).
  - To modify the values of "User Input Fields" in real-time (new).

  This is supported for normal jobs as well as for cluster jobs. Any manual actions made in this overlay are journaled in the test result (*.prxres file) and can be retraced in the "Load Test Result Detail Menu" → "Test Scenario".

  Notes: a) If a load test job is suspended it will nevertheless be terminated when the planned test duration is exceeded.
       b) If you decrease the number of users to zero then the load test job will stop as soon as zero users are reached.
       c) Only already running users can be decreased – decreasing of users which are started in future is not supported.
       d) If you trigger an increase or a decrease during an already increasing/decreasing process then a second increasing/decreasing process will run in parallel.
       e) User Input Fields can only modified in real-time if they have been created/declared by enabling the (new) checkbox "Enable Value Changes in Real-Time - During Test Execution".

- **Reduced Overhead Between Executed Loops**
  The overhead of a couple of 10 Milliseconds between the executed loops of a simulated user has been reduced to almost zero.

- **Reduced File Size / Memory Size for Cluster Job Results**
  When acquiring the result of a cluster job you can now select if all of the "raw" Exec Agent member data should be part of the test result (as before). If you de-select this (new) option then the size of the result file – respectively the size of the required Java memory to display the result – will be up to 90% smaller – depending on the number of cluster members. You will not lose any data – but the cluster job result cannot be expanded later into singe Exec Agent job results.

- **Optimized Resources for "One Loop Only per User" Load Tests**
  The Java memory and the references to terminated threads has been substantially reduced for "One Loop Only per User" load tests

- **Measuring of the SSL Handshake Time**
  The time elapsed to perform SSL Handshakes is now measured and displayed in real-time during the test execution as well as in the load test results.

- **Removed Load Test Options**
  The load test options "-log", "-nolog" and "-sslcreset" have been removed. Passing such options take now no effect. The SSL session cache is now cleared at the start of each executed loop (as by setting "-sslcreset" before).

- **New PrxJob Commands**
  The following PrxJob Commands have been added**:**

  - changeJobNumSimulatedUser   \<exec agent name> \<job id> \<delta number> [\<startup delay ms>]
  - setJobSuspend   \<exec agent name> \<job id> \<"false"|"true"> [\<startup delay ms>]
  - isJobSuspend   \<exec agent name> \<job id>
  - changeJobTestDuration   \<exec agent name> \<job id> \<delta seconds>
  - getJobRealTimeUserInputFields   \<exec agent name> \<job id>
  - getClusterJobRealTimeData   \<cluster name> \<cluster job id> [-detailed]
  - changeClusterJobNumSimulatedUser   \<cluster name> \<cluster job id> \<delta number> [\<startup delay ms>]
  - setClusterJobSuspend   \<cluster name> \<cluster job id> \<"false"|"true"> [\<startup delay ms>]
  - isClusterJobSuspend   \<cluster name> \<cluster job id>
  - changeClusterJobTestDuration   \<cluster name> \<cluster job id> \<delta seconds>
  - getClusterJobRealTimeUserInputFields   \<cluster name> \<cluster job id>

- **Revised PrxJob checkUrl Command**
  The PrxJob checkUrl Command supports now SNI (Server Name Indication). Furthermore, the following new options have been added:
  - o [-ssl \<version>]   (Optional. Set the SSL or TLS version. Possible values are "all", "v3", "tls", "tls11" or "tls12")
  - o [-dnsstatistic]   (Optional. Measure also the DNS resolve time.)
  - o [-dnssrv \<IP-name-server-1>[,\<IP-name-server-N>]]   (Optional. Set an alternative list of DNS servers.)

- **Two New Internal Processing States for Measured URL Calls**
  The internal processing states "DNS Resolve" and "SSL/TLS Handshake" have been implemented for measured URL calls. In case if an URL call fails (for example when a timeout occurs), you will see now if such an error did potentially happen during query of a DNS server or during a SSL/TLS handshake.

  Complete list of all internal processing states:

Internal Processing State of URL Call	Value	Meaning
No Step / Not Initialized	-1	The URL call had not yet started.
DNS Resolve (new)	10	The URL call failed during resolving the DNS name of the web server.
Open Network Connection to Proxy	0	The URL call failed during the opening of a network connection to an outbound proxy server. This processing state is executed only if the load test is performed via a proxy server (unusual, not recommended test scenario).
Open Network Connection	1	The URL call failed during the opening of a TCP network connection to the web server.
SSL/TLS Handshake (new)	11	The URL call failed during executing the SSL/TLS handshake with the web server.
Transmit HTTP Request	2	The URL call failed during the transmission of the HTTP request data.

     All Rights Reserved

Wait for Server Response	3	The URL call failed while waiting for the first byte of the HTTP response data from the web server.
Receive HTTP Header	4	The URL call failed while receiving the HTTP response header from the web server.
Receive Content	5	The URL call failed while receiving the HTTP response content from the web server (HTML data, images, ...).
Close Network Connection	6	The URL call failed while closing the network connection to the web server.
All Done	7	The URL call itself completed successfully (all data transmitted and received), but the received HTTP status code was incorrect, or the received MIME type (text/html, image/gif, ...) was incorrect, or an error was detected inside the received content data.

- **DNS Resolve Time and SSL Handshake Time Shown in URL Performance Data**
  The DNS Resolve Time and the SSL Handshake Time are now shown in all URL performance data.

- **Support to Terminate a Simulated User due to a Failed URL Call**
  The "Failure Action" in the "HTTP Response Verification" menu contains a new option which allows to terminate a simulated user when a "red error" occurred for the (particular) configured URL. All simulated users for which this error occurs are removed from further load test execution. The users are also terminated if a variable extraction for that URL fails or when a load test plug-in which is bound to that URL releases a self-generated error. The corresponding captured error snapshots are marked with a yellow colored warning sign in order that you can retrace why a user has been terminated.

- **Accessible X509 Server Certificate Chain**
  The X509 certificate chain received from the web server during the SSL/TLS handshake is now accessible from load test plug-ins. This allows also to extract the public key of a Web server certificate during the load test execution.

- **New Functions added to the Proxy Recorder REST API**
  The following functions have been added to the Proxy Recorder REST API:
    - getRecorderURLBlacklist
    - setRecorderURLBlacklist
    - getRecorderURLWhitelist
    - setRecorderURLWhitelist
    - getRecorderOutboundProxyTimeout (Apica BNet only)
    - setRecorderOutboundProxyTimeout (Apica BNet only)
    - getRecorderRecordFailedTransmitRequests (Apica BNet only)
    - setRecorderRecordFailedTransmitRequests (Apica BNet only)
    - getRecorderSSLVersion

o setRecorderSSLVersion

The updated Application Reference Manual contains further information about these new REST API functions.

- **Additional Java Methods for Recorded Sessions and Recorder Plug-Ins**
  - o Class dfischer.proxysniffer.ProxyDataRecord
    - New method getRecorderSequenceID()
    - New method getExecutionStep()
    - New method getFailureExceptionText()

- **Additional Java Methods for Recorder Plug-Ins**
  - o Class dfischer.proxysniffer.ProxyRecorderContext
    - getRecorderSequenceID()
  - o Class dfischer.proxysniffer.ProxyRecorderPluginURLContext
    - getRecorderSequenceID()
    - getExecutionStep()
    - getFailureException()

- **Revised Installation Kits**
  The installation kits for Windows, Mac OS X and Linux have been revised. The Mac OS X installation kit contains now an integrated Java compiler and interpreter.

2.1.2 Minor Release V5.4-D (Installation Kit dated October 8, 2015)

ZebraTester V5.4-D contains the following new features:

- **Support for Log4j**
  As an option, all components of ZebraTester (but not the executed load test programs) can configured in such a way that log4j is used for logging. The startup option -log4j has added to ProxySniffer, WebAdmin, ExecAgent and JobController. For further information see the revised Application Reference Manual.

- **New Functions added to the Proxy Recorder REST API**
  The following functions have been added to the Proxy Recorder REST API:
  - getRecorderURLRegexBlacklist
  - setRecorderURLRegexBlacklist
  - getRecorderURLRegexWhitelist
  - setRecorderURLRegexWhitelist

  The revised Application Reference Manual contains further information about these new REST API functions.

- **New Function added to the PrxJob Utility**
  The following function has been added to the PrxJob Utility:
  - getClusterJobStartStatistics

  The revised Application Reference Manual contains further information about this new PrxJob function.

- **Enhanced Support of the Apica Load Test Portal (LTP)**
  The functions to upload and download of files to/from the Apica Load Test Portal (LTP) have been enhanced and support now all new features of the LTP.

- **Signed Firefox Recording Extension**
  The Firefox Recording Extension has been revised and signed and supports now the latest Firefox versions.

- **Upgrade to Java 7**
  ZebraTester requires now Java 7 (instead of Java 6). The Windows and the Mac OS X installation kits contain now an integrated Java 7 compiler and interpreter.

- **Improved Mac OS X Installation Kit**
  The installation of ZebraTester on Mac OS X has been simplified and the Mac installation kit supports now the latest Mac OS X versions "Yosemite" and "El Capitan".

2.1.3   Minor Release V5.4-I (Installation Kit dated March 8, 2016)

ZebraTester V5.4-I contains the following new features:

- **GitHub Integration of the MyTests Directory Tree**
  By using the GitHub integration you can manage all of your recorded sessions and all of your load test result with a remote revision control system. This allows you to have multiple branches of the MyTests directory tree, and to track changes/modifications of any file. Furthermore the content of old, overwritten files can also restored. The GitHub integration is accessible from the Project Navigator menu. For further information see the new **GitHub Integration Manual**.

- **New Feature: "User-Defined" Transactions**
  By defining "User-Defined" Transactions in the GUI you can mark sections in the recorded session which can start at any URL and can end at any succeeding URL or page. Later, when executing the load test, performance data of such sections are separately measured, which means that the "transaction time" and the "number of transactions" are captured. For example: If you define a transaction around a login procedure, you can measure how many logins per second have been performed during the load test. The measured data of transactions are shown in real time while executing the load test, and are shown in the load test result details. In addition, the measured data of transactions are also shown as "load curve diagrams".

- **New Feature: Inline Scripts**
  Inline Scripts allow you to enhance the load test program with additional features. Like for example: Modify the HTTP request data before transmitting an URL call, or extracting data fragments from HTTP responses into variables. Similar as you can do with Load Test Plug-Ins - but in an (much) easier way. Inline Scripts can also abort the current executed loop of a user and trigger a "red" URL error snapshot.

  Inline Scripts are written in **Prx BASIC** programming language. The runtime behavior of Inline Scripts can be tested in advance, whereby the script environment is simulated based on the data of the recorded session. For full documentation see **Prx BASIC Language Reference Manual**.

- **Support of EC2 Data Center "Seoul"**
  AMIs for load generators and "Pure Cloud" instances are now also available in the EC2 Data Center "Seoul".

- **Proxy Recorder Certificates now with SHA-2 signed**
  The Web server certificates created by the Proxy Recorder are now signed using the SHA-2 algorithm (instead of using SHA-1 as before). The updated **CreateOwnCARootCertificate** tool delivered by the new installation kits supports also to create an own Root Certificate signed by the SHA-2 algorithm.

  Note in case if you are upgrading ZebraTester from an older version: Because some Web browsers inclusive Firefox (on all OS) and also Mac OS X and iOS 9 don't longer accept certificates signed by the SHA-1 algorithm we recommend that you delete/remove the old ZebraTester CA root certificate from your Web browser and from your OS. After that create a new one by using the updated **CreateOwnCARootCertificate** tool provided by the installation kit (see ZebraTester Installation Guide) and then install your new CA root certificate into the Web browser and import it into the OS.

  By clicking on the certificate icon in the ZebraTester main menu you can verify that the new root certificate is now signed by the

"sha256WithRSAEncryption" algorithm (don't confuse it with the certificate fingerprint which is still generated in standard form using SHA-1). Finally you should delete all old Web server certificates by using the Web Admin GUI in order that they are newly created with the SHA-2 algorithm. After that restart the ZebraTester Console.

- **New Diagram: SSL Handshake Time**
  The captured SSL Handshake Times are now shown as a time/value diagram in the "Load Test Result Detail" menu.

- **New Diagram: Load Generator Performance**
  The CPU Usage of the load generator(s) and the "Java Garbage Collector Rate" of the load test jobs is now measured during the load test execution and added to the load test result. Both data collections are shown as time/value diagram in the "Load Test Result Detail" menu.

- **Job ID and Job Max. Java Memory added to Test Scenario/Objectives**
  The Job ID and the Job Max. Java Memory (max. available Java memory of load test job) has added to the "Load Test Result Detail → Test Scenario/Objectives" menu.

- **Optimized PrxCombineTestResults Tool**
  The PrxCombineTestResults command line tool has been optimized and does no longer append by default the discrete results of the cluster members to the combined data. You can re-enable the old behavior by passing the new argument -addDiscreteResults .
  This tool is now documented in the revised Application Reference Manual.

- **Enhanced PageScannerSessionToJava Tool**
  The PageScannerSessionToJava command line tool – which allows you to convert a recorded session into a Java load test program – supports now the optional arguments -noStripReferer and -noStripAccept. Further information is available in the revised Application Reference Manual.

2.2 Discontinued Functions

- The support for "SSL V2" has been removed
- Kerberos authentication has been removed

2.3 Version 5.4 Compatibility

Previous Version Data Files / Version 4.6-Z – 5.2-*	Compatibility with Version 5.4
Recorded Web Surfing Sessions (*.prxdat files)	Compatible
Created Load Test Programs (*.java and *.class files)	Compatible
Load Test Results (*.prxres files)	Compatible

Previous Version Data Files / Version 4.0 – Version 4.6-Y	Compatibility with Version 5.4	
Recorded Web Surfing Sessions (*.prxdat files)	Compatible	[1] Workaround: you can load any old web surfing session (*.prxdat file) into V5.4 by using the Project Navigator. After that you have to generate and compile the load test program again before you can run it on V5.4.
Created Load Test Programs (*.java and *.class files)	NOT Compatible [1]	
Load Test Results (*.prxres files)	Compatible	