

# SuperZap User Guide V1.0.09 by Automated Software Tools

## Table of Contents

1	Introduction
2	Installation
3	Execution
3.1	Standalone graphical user interface mode
3.2	Command line interface mode
3.3	Batch command file interface mode
3.4	Web browser graphical user interface mode
4	Commands
5	Error messages
6	Examples
7	References
8	Appendix
8.1	Command Reference Table
8.2	Error Message Reference Table
8.3	Examples
8.3.1	demo1.zap - dump file in hex, ASCII, and EBCDIC
8.3.2	demo2.zap – find, verify, and replace
8.3.3	demo3.zap - create a test file using macro instruction loop
8.3.4	demo4.zap – create a test file in ASCII and EBCDIC and find record
8.3.5	demo5.zap – create a test file using subroutine and format report with totals
8.3.6	demo6.zap – generate a file containing prime numbers
8.3.7	demo7.zap – monitor status of real time file updates in a GUI window
8.3.8	demo8.zap – resize scrollable GUI window and font size for usability
8.3.9	demo9.zap – show pc processor benchmark statistics using status line
8.3.10	demo10.zap – interactive file query using GUI, command line, or CMD
8.4	Release Notes

## 1.0 Introduction

The SuperZap file utility for Windows provides a fast and easy way to display and/or change the contents of any Windows file up to 2 GB in size. A unique time stamped log file is created to document all of the changes made to a file. The SuperZap file utility basic commands OPEN, DUMP, VER, and REP perform the functions of opening, displaying, verifying, and replacing file content respectively. If the verify commands are not successful then the following replace commands are suppressed.

The SuperZap file utility can be executed as a standalone graphical interactive application, a command line interactive application, a batch command, or as a web based interactive application. The graphical user interface (GUI) conforms to industry standards including:

- Popup menus – file, edit, view, actions, and help
- Scrollable view of entire log showing results of each command

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

- Command entry field with keyboard scrolling of prior commands
- Optional toolbar with combo box to select file action commands
- Optional status line to show instructions, I/O, and mode status information
- Option to change window location, size, and font via commands
- Option to execute command macro files with parameter substitution
- Option to execute interactive Windows commands using GUI interface

The menus and basic commands all have mnemonic keystroke equivalents. All fields including the entire log view have edit cut, copy, and paste support to allow easy transfer of selected text to and from the clipboard and other Windows applications. The main window is resizable and the toolbar and status line can be hidden to allow viewing multiple files concurrently with multiple copies of SuperZap file utility.

In addition to the interactive GUI interface, SuperZap supports execution from interactive command line or a batch command file. The command files can pass user supplied parameters and can invoke nested SuperZap command files.

There is a set of SuperZap macro command language instructions for use in defining macro command files that can perform functions with program control such as the following:

- Search for selected hex or text data
- Format reports to display selected fields or calculate statistics
- Dump or display data in various formats referencing data fields using relative byte addressing from current position in file:
  - Hex with or without ASCII or EBCDIC text
  - Text only with optional filter to remove noise
  - Format integer fields using either Big Endean or Little Endean format
  - Format packed decimal
- Update data fields using hex or text data from commands, user parms, or input from console or data file via ACCEPT command.
- Monitor online file and display current status based on file changes.

The SuperZap program itself is written in Java and requires that the free Java 2 J2SE runtime 1.4+ from Sun Microsystems with the Swing GUI support installed on the Windows PC. The same superzap.jar Java code can be executed as a web based Java Applet application, Windows standard GUI application, command line application, or batch command file application.

The SuperZap file utility can be executed directly as a Java Applet by clicking on the SuperZap icon on the following web site:

<http://www.SuperZap.net>

The web browser based version will default to demo mode if Java security manager file access permissions are not defined. Demo mode uses a dummy file consisting of

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## **SuperZap User Guide V1.0.09 by Automated Software Tools**

**256 bytes containing the hex values hex 00 to hex FF. The dummy file is opened when any OPEN or NAME command is issued using file menu or command line entry. It is possible to define a Java Security Manager policy to grant file access rights to the web based SuperZap file utility which then allows it to perform all functions on any file. The open source General Public License version of the SuperZap file utility for windows can be downloaded and installed from the same website. The download file consists of an InstallShield executable file. Once installed, the licensed version will have access to all functions on any Windows file up to 2 GB with no time restrictions. Users who find the utility useful and wish to support continued development and support of open source compliant utility software, can support the continuing effort by making an online donation.**

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 2.0 Installation

To download the demo version, go to the following website and click on download link:

<http://www.SuperZap.net>

After downloading the desired zip file superzap.zip, unzip the file and execute the Install Shield superzap.exe program to install the SuperZap utility. The default install directory for Windows is:

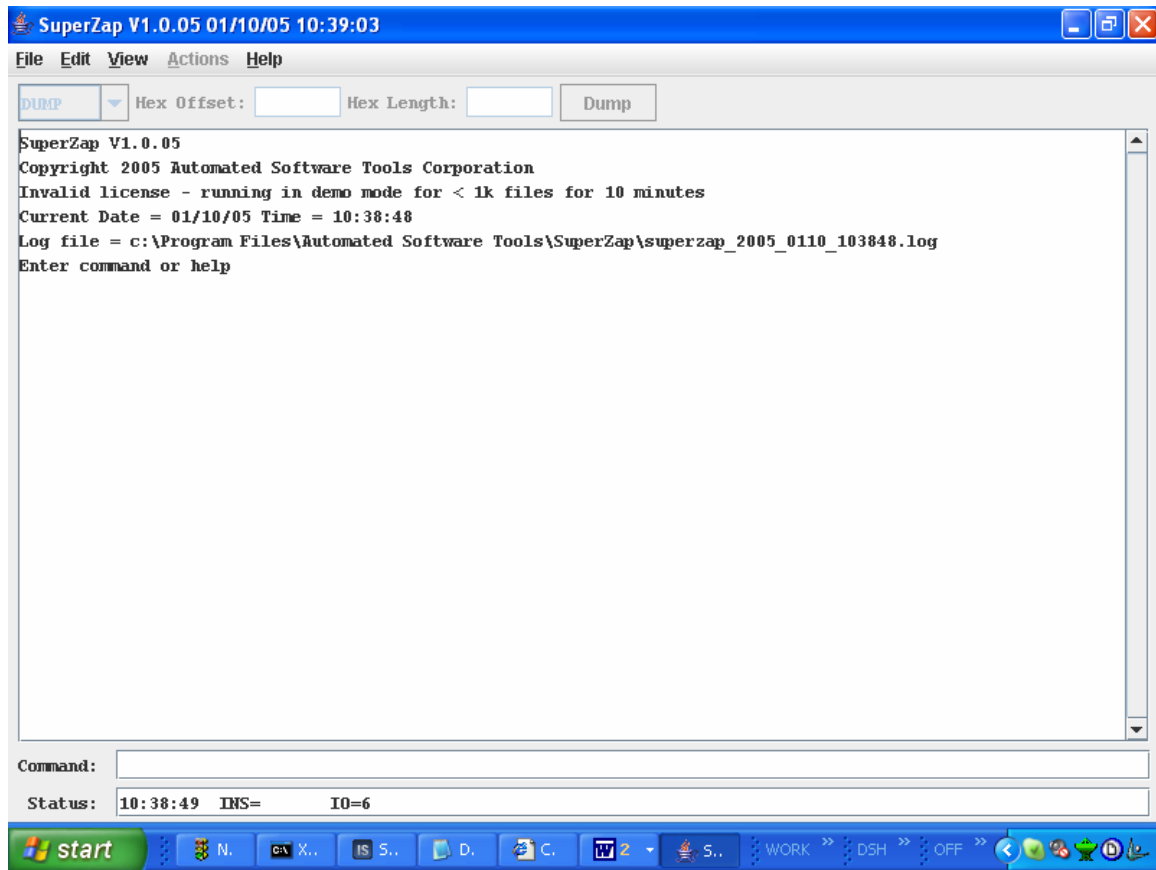
“c:\program files\Automated Software Tools\SuperZap\”

If you change the install location, the Windows command line file ZAP.BAT used to launch SuperZap GUI or command line execution of zap macro files will need to be changed according. This zap.bat file can be added to system path in order to be able to run superzap from any Windows command line directory. In addition, the file type .zap can be added to Windows file types to allow double clicking on any .zap file to invoke SuperZap.

Once the shareware SuperZap file utility for Windows is successfully installed, it will support all functions on any Windows file up to 2 GB with no time restrictions.

## SuperZap User Guide V1.0.09 by Automated Software Tools

The install will place the SuperZap icon on the desktop for invocation as a standalone Windows GUI application which looks like this:



If you have not already installed the free version of Sun Microsystems Java 2 J2SE runtime 1.4.2+, it can be downloaded and installed from the following web site (go to website and click on “Download JRE”):

<http://java.sun.com/j2se/1.5.0/download.html>

You do not need the Sun Microsystems Java SDK System Development Kit or the J2EE Enterprise Edition components which also include the same runtime. The runtime install download file for J2SE 1.5.0 runtime for Windows is about 14 MB and has the name: jre-1\_5\_0-windows-i586.exe

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 3.0 Execution

Once the installation is complete, the SuperZap file utility for Windows can be executed in any one of the following modes:

- **3.1 Graphical User Interface (GUI) application mode**
  - Double click on the SuperZap file utility icon on the desktop to startup in the standalone graphical user interface mode.
  - From a command line, the graphical user interface can be started using the batch command file zap.bat located in the demos directory. Use the command: zap /g or simply zap.
- **3.2 Interactive command line mode**
  - From command line use command: zap /c.
- **3.3 Batch execution mode**
  - From command line you can start a zap command file in batch mode such as demo1.zap using the command: zap demo1
- **3.4 Web browser Java Applet GUI application mode**

You can start web browser page with the following html code which will display the SuperZap icon which will start GUI application when icon is clicked with mouse:

```
<html>
<body><p>Test superzap applet launcher</p>
<table><TR><TD>
<applet code=superzap archive=superzap.jar width=48
height=48></applet>
</TD></tr></table>
</body>
</html>
```

- Note that running SuperZap GUI from web browser will result in the log file being placed on the desktop. It can be moved or deleted after execution. By default the log will have unique time stamp included. With option /nt specified, the name will be superzap.log.

This html code is enabled on the [www.SuperZap.net](http://www.SuperZap.net) web site to allow running in demo mode. Note that web browsers typically disable all Java security permissions as the default so only the demo functions will be available in this mode unless the Java PolicyTool is used to enable permissions for this program. See the security page on [www.SuperZap.net](http://www.SuperZap.net) for example of how to set Java security permissions using PolicyTool.

## SuperZap User Guide V1.0.09 by Automated Software Tools

The following additional parameters can be passed to superzap on any of the above startup commands:

- **/NP** – suppress all permissions to test web version
- **/NT** – suppress the unique time stamp included in the default log file name (By default the unique time stamp insures a log of all SuperZap commands executed are retained for each instance of SuperZap GUI window, command line, and batch file executions. The log files can be deleted periodically.)
- **/RT** – regression test suppression of date and time stamps to simplify file comparisons
- **/SC <file>** - startup bat command file to start Windows command after switching to interactive Windows command mode when running in GUI. This allows front ending any existing Windows interactive command line application with GUI front end supporting scrollable output window etc.
- **/SZ <file>** – startup zap command file to set user options such as location and size of window, font size, change directory, hide toolbar, ASCII or EBCDIC, dump display size options, big Endean or little Endean integer display format, etc. This command file can also include application commands such as changing options for the GUI window location, size, font, toolbar, status line, etc. It can also open a file and format the display of selected information. For example, you could open a small one line display window in one corner of your window that display control information from an online application shared file via SuperZap or DOS commands. See demo7 for an example.

To eliminate the need to specify the full path for the SuperZap program itself for each execution, the path can be added to the system path by going to Windows control panel, starting the “System” dialog, selecting “Advanced tab” and then “Environment Variables” button. Select the variable named “PATH”, and add the following path to the end:

**C:\program files\automated software tools\superzap**

### 4.0 Commands

See Appendix I for table with all of the SuperZap basic commands. This table contains the following columns:

- **Command**
  - Commands can be upper or lower case
  - Commands shown in lower case are only available via GUI menu
- **Parameter 1**
  - All parms starting with 0-F are assumed to be in hex
  - All parms starting with Rn are assumed to reference registers where n is the number of the register 0 - F

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

- Hex relative byte address offsets may contain reference to current file storage (FS:) or working storage (WS:) relative byte address position using asterisk as well as register and hex references. For example:
  - 1C references file offset 28 decimal
  - \*-4 references the current file RBA offset minus 4
  - \*+R5-2 references the current file RBA plus contents of R5 minus 2
  - Entering a value in this field and pressing enter will update the current RBA address to the specified calculated address. Then following commands can reference that RBA using \*.
- Parameter 2
  - Parameter 2 can be separated from parameter 1 by comma or spaces
  - Parameter 2 may contain multiple operands separated by commas such as 'text',0d0a. Text is entered in single or double quotes and generates hex bytes based on current ASCII or EBCDIC setting.
- GUI Menu containing the command if any
- Alternate keyboard entry for command if any
- Description of the command

The following mode settings affect how some commands function:

- **CMD** command switches to interactive Windows command mode during which all input commands are routed to Windows command processor and output from Windows command are directed to log. Use CTRL-BREAK or F3 to exit interactive CMD mode. Individual Windows commands which do not prompt for additional input can be executed without switching to CMD mode by prefixing them with CMD. If a Windows command does not conflict with SuperZap commands or command files, then they can be entered without the CMD prefix. See demo2.zap for example use of Windows erase command within a zap file. See demo10 for example of how an interactive command line query application can be run as Windows command with SuperZap GUI front end to provide scrolling out view, optional window and font sizes, etc.
- The default character set mode is ASCII. The character set mode can be changed to EBCDIC via the view menu or using EBCDIC command. The character mode determines how hex bytes in a file will be interpreted as text in dump and show commands and how literal text in commands will be converted to hex.
- The default dump view mode is to dump entire file (VIEWF) provided the current file is less than 1 million bytes. This allows easy scrolling of entire file to view any section. If the file is over 1 million bytes, the default view is changed to view page (VIEWP). There is also optional view line (VIEWL) mode where each dump command shows next line. Note that in page and line view mode, the dump commands advance the current relative byte address in the file.
- The default integer storage mode is Little Endian (LITTLE) which means the low order byte of integers is stored in lowest storage address. The mode

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

can be switched to Big Endian (BIG) which means low order byte of integers is stored in highest storage address. LITTLE is native Intel format, and BIG is native IBM mainframe format.

- The default offset and integer value base is hex 16 (HEX) for compatibility with the SuperZap VER and REP commands. The default base can be changed to decimal 10 (DEC). Note that when using decimal mode, literal byte data is still specified in hex. For example, in DEC mode, MVI 10,1020 means move the two hex bytes 1020 to decimal offset 10. See demo5 for example use of DEC mode versus the default hex mode for integers and offsets used in demo3.
- The default storage mode is file storage (USING FS). Once a file is opened, instruction offset references refer to relative byte address within the open file. Offset references up to the current length of the file are valid. The current relative byte address (RBA) is initially zero and can change as result of accept, dump, display, find, or rba commands. The current file storage rba is displayed on the status line while file is open. The default storage mode can be changed to working storage (USING WS). Working storage offsets are valid from 0 to current length which defaults to 4096. The length of working storage can be changed using SWL command. References to file storage or working storage can be made in any instruction while in either mode using override prefix of FS: or WS:. These overrides can also be used in defining symbolic references via the EQU command. See demo5 for example of symbol record layout and use of storage override prefixes. The file storage is buffered through a 4096 byte buffer which is only read or written when the next relative byte address is outside the current buffer. For this reason, performance of file storage is a function of how much random access is done over 4096 bytes apart where as working storage is in real memory and generates no disk I/O other than paging if the system is over committed on real memory.

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 5.0 Error messages

Append II contains table with all the SuperZap file utility error messages along with additional information on problem determination and resolution.

Registered users may also request help via email to:

[support@AutomatedSoftwareTools.com](mailto:support@AutomatedSoftwareTools.com).

SuperZap error messages are preceded by the text “SuperZap Error S???” so any log file can be quickly searched for error messages.

One of the most powerful tools for debugging zap command files is to use the TRACE ALL command which displays optimized command line for each instruction, register contents, and PSW including condition code, file rba, and working storage rba after each command. You can insert TRACE ALL at beginning of zap file, run the zap file, and then view the generated log file. When debugging, you may find it helpful to add the /NT option to remove time stamp from log. For example the command zap demo1 /nt will then generate demo1.log file. I do not recommend using /NT option in command files for production use as it opens the door for lost log information regarding file updates.

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 6.0 Examples

The best way to learn how to use the SuperZap file utility is to look at some examples and try out the GUI interface. Then you can refer to the command reference table in Appendix I for help as required: Appendix III contains the following examples:

1. demo1.zap - dump file in hex, ASCII, and EBCDIC
2. demo2.zap – find, verify, and replace
3. demo3.zap - create a test file using macro instruction loop
4. demo4.zap – create a test file in ASCII and EBCDIC and find record
5. demo5.zap – create a test file using subroutine and format report with totals
6. demo6.zap – generate a file containing prime numbers
7. demo7.zap – monitor status of real time file updates in a GUI window
8. demo8.zap – resize scrollable GUI window and font size for usability
9. demo9.zap – show pc processor benchmark statistics using status line
10. demo10.zap – show interactive file query using GUI, command line, or CMD

Once you have installed SuperZap, you can start the GUI interface by double click on the SuperZap desktop icon. Change directory to the demos directory by typing `cd demos` and then type any one of the 10 demo file names above to run the demo and see the generated output. There is also a Windows batch file named `demos.bat` in the demo directory that can be used to run all 10 demos in GUI mode using the following 3 commands to drive Windows process from GUI:

1. CMD
2. CD DEMOS
3. DEMOS.BAT

Here are some examples for the most common functions:

- Using just a mouse and the GUI menus
  - Open a file
    - Select “file” menu and select “Open” which will bring up file selection dialog. Position to the file desired and click open.
    - It is not necessary to save or close any open file as it is done automatically when opening another file or exiting the program.
  - Dump a file
    - Select “Action” menu and select “Dump” for hex and text dump using the current dump file, page, or line option. You can scroll the log view to see any part of the dump.
    - You can also select any of the other dump options “DumpH” for just hex (32 bytes per line), “DumpT” for printable text (64 bytes per line), “ShowT” for all printable text only, or “ShowW” for only printable text between 3 to 80 contiguous bytes.

## SuperZap User Guide V1.0.09 by Automated Software Tools

- You can change dump options on the “View” menu including:
  - ASCII or EBCDIC printable character display
  - File, visible page, or single line dump option. The default is to dump entire file unless it’s over 1 M
- Using the toolbar to dump selected section of a file
  - Select the type of dump by clicking on command combo box
  - Enter the hex relative byte address (RBA) in the first field.
  - Enter the hex length in the second field and press enter or click on the action button.
  - The toolbar also supports the following additional commands:
    - VER – verify hex and/or text data
      - Set first field to desired starting RBA
      - Set second field to one or more parameters separated by commas
        - Hex data
        - ‘text’ character data which is converted to hex using current ASCII or EBCDIC setting.
      - Press enter at end of second field or click on action button
    - REP – replace hex and/or text data
      - Set second field to one or more parameters separated by commas
        - Hex data
        - ‘text’ character data which is converted to hex using current ASCII or EBCDIC setting.
      - Press enter at end of second field or click on action button
    - Display – format text and data into a display line
      - Change between file storage or working storage if necessary using the USING FS/WS command. The default is file storage
      - Set initial file or working storage offset as the first optional parameter of DISPLAY statement, if omitted it will use the current RBA setting.
      - Set second field to format parameters as follows:
        - ‘text’ - literal display text
        - Hn - hex display of next n bytes
        - In - decimal display of integer field of length n using the current default LITTLE Endean or BIG Endean format. You can change the default setting on the “View” menu.
        - Pn – decimal display of packed decimal field of length n

## SuperZap User Guide V1.0.09 by Automated Software Tools

- Tn – text display of next n bytes using the current ASCII or EBCDIC text setting.
    - The toolbar is not required and can be hidden along with the status line to save window space by selecting the “View” menu and deselecting the “Toolbar” and/or “Status line” items.
  - Use the command line
    - Any SuperZap command, command file, or Windows command can be entered from the command line
      - The command can be upper or lower case and can start anywhere
      - Optional operands can follow
      - Comments can follow operands and if there are no operands a “,” must precede the comments
      - Previously entered commands can be viewed using the up and down arrows while focus is on the command line
      - If a command is not recognized as a SuperZap command, then it will search for zap command file by that name and execute it with parameter substitution for each occurrence of %1 through %9 in the zap command file. If no zap command file is found, then the command is prefixed with “CMD “to attempt executing it as a native Windows command.
      - For native Windows commands that are duplicates of SuperZap commands or zap command files, prefix the command with “CMD “.
      - Alternatively, the command CMD by itself will switch the command line to native Windows command mode, and all input will be routed to Windows command processor for interactive Windows commands. To cancel the Windows interactive mode, use CTRL-BREAK key or F3. These keys will cancel CMD mode first, zap mode second, and entire application if neither mode is active. Note that in command line mode CTRL-C is used to cancel application whereas in GUI mode CTRL-C is reserved for copy accelerator key.

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 9 References:

- 1 [IBM OS/390 Diagnosis: Tools and Service Aids](#) (See AMASPZAP)
- 2 [Sun Micro Systems Java 2 J2SE Runtime](#)
- 3 [Microsoft Windows](#)

# SuperZap User Guide V1.0.09 by Automated Software Tools

## Appendix I: SuperZap Commands in alphabetical order:

SuperZap Commands for Version 1.0 as of 01/04/05

	Command	Parm 1	Parm 2	Description
1	ABOUT			Display summary information about SuperZap file utility for Windows
2	ACCEPT	o1	l1	Accept input from command line or text file opened via OPENA into file (fs) or working storage (ws) at offset. Default length is 80 if not specified. Data will be set to current ASCII or EBCDIC text. Note ACCEPT sets fs/ws rba to current record location read.
3	ACTR	<hex>		Set instruction loop limit. Default is 4096
4	AR	r1	r2/i2	Add r2/i2 to r1
5	ASCII			Set display text mode to ASCII code set (default)
6	B	label		branch to label
7	BASE	<hex offset>		Set base offset to be added to hex offset references
8				branch to label if psw condition code set to Equal
	BE	label		
9	BH	label		branch to label if psw condition code set to High
10	BIG			Set default integer storage to big Endean for compatibility with mainframe files. Default is little Endean (LITTLE)
11	BL	label		branch to label if psw condition code set to Low
12	BNE	label		branch to label if psw condition code not set to Equal
13	BNH	label		branch to label if psw condition code not set to High
14	BNL	label		branch to label if psw condition code not set to Low
15	BNO	label		branch to label if psw condition code not set to Ones
16	BNZ	label		branch to label if psw condition code not set to Equal
17	BO	label		branch to label if psw condition code set to Ones
18	BZ	label		branch to label if psw condition code set to Equal
19	C	r1	o2	compare register r1 to 4 bytes at fs/ws offset o2

## SuperZap User Guide V1.0.09 by Automated Software Tools

20	CD	<directory>			Change current directory to allow entering just the file name in following OPEN/NAME commands. CD is not needed if you use the OPEN select file dialog.
21	CH	r1		o2	compare register r1 to 2 bytes at offset o2
22	CLC	o1		l2	compare field at fs/ws offset o1 for length l2 with current ws rba offset
23	CLI	o1		i2	compare bytes at fs/ws offset o2 to immediate hex/text data i2
24	CLOSE				Close the current open data file after updating. This command is disabled on the file menu if no data file is open, and file is updated and closed at exit automatically
25	CLOSEA				close currently open ACCEPT file opened via OPENA
26	CMD	command			Switch to Windows command input mode if no command specified, else execute specified Windows command. See TIMEOUT for time limit control of Windows commands. Use EXIT, or CNTL-BREAK to cancel command in process.
27	Copy				Copy selected text in GUI window field to the clipboard
28	COMMANDS				Alphabetical list of all SuperZap commands with operands
29	CONSOLE	ON/OFF			Turn on console output of log in batch mode
30	COPYLOG				Copy the entire log text to the clipboard
31	CR	r1		r2/i2	compare r1 to r2/i2
32	Cut				cut selected text in GUI window field
33	CVB	r1		o2	convert packed decimal bytes at fs/ws offset o2 to r1. Field length is determined by FIELD command (default 8).
34	CVD	r1		o2	convert r1 to packed decimal bytes at fs/ws offset o2. Field length is determined by FIELD command (default 8).
35	DEC				set instruction offset and integer value base to decimal 10 versus hex 16 default. Note immediate hex data can still be entered in hex.

## SuperZap User Guide V1.0.09 by Automated Software Tools

36	DISPLAY	o1	text',Hn,In,Pn,Tn	Format output to log starting at fs/ws offset specified or default to current hex offset. The format is controlled by one or more 'text' strings and data format types as follows: Hn for hex, In for integer, Pn for packed decimal, Rn for integer value of register, and Tn for text data. Note display updates fs/ws rba for each field formatted.
37	DR	r1	r2/i2	divide r1 by r2/i2
38	DUMP	o1	l2	Dump showing hex and text (no parms to dump entire file). Note dump updates fs rba in page or line mode.
39	DUMPH	o1	l2	Dump showing hex only (no parms to dump entire file)
40	DUMPR	r1	r2	dump specified registers or all registers
41	DUMPT	o1	l2	Dump showing text only (no parms to dump entire file)
42	EBCDIC			Set display text mode to EBCDIC code set
43	ECHO	ON/OFF		turn command echo to log on or off (default is on)
44	EQU	value		Define symbol for ws/fs offset or register (may include ws: or fs: prefix for offset to override current fs/ws rba setting. Symbol value may reference any previously defined symbols using + or - operators. For example key equ record+10. Symbols may be used in place of offsets or registers. See demo5 for examples.
45	EXIT			Cancel previous CMD if still running, else cancel current ZAP file execution if any. If no CMD or ZAP active, then exit SuperZap after updating and closing all files. Note CTRL-C can be used in place of EXIT when running in command mode. There is no accelerator for EXIT in GUI mode as CTRL-C is reserved for COPY.
46	FIELD	hex		set default field width for SHOW, CVB, CVD. Default is 8
47	FIND	o1	i2	Find hex or text starting at specified offset. If hex/text key is omitted, the current work area will be used as key. Note FIND updates fs rba only if key found. FIND sets PSW condition code to E if found else L.
48	FONT	size		set font size (default is 12)
49	GUIDE			Display user guide manual in IE browser
50	HELP			Display summary of commands

## SuperZap User Guide V1.0.09 by Automated Software Tools

51	HEX			set default operand base to hex 16 (see DEC for decimal)
52	IC	r1	o2	insert character from file offset o2 into register r1
53	L	r1	o2	load register r1 from 4 bytes at file offset o2
54	LFL	r1		set r1 to current open file length. Set to -1 if file not open
55	LH	r1	o2	load register r1 from 2 bytes at file offset o2
56	LIMIT	l1		Limit find search (default is entire file length)
57	LITTLE			Set default integer storage to little Endean format for compatibility with Intel pc format. (default)
58	LOC	x	y	Set location of main GUI window in decimal pixels. Upper left corner of screen is 0,0. Location limited to insure at window is visible. See SIZE command also.
59	LOG	<file>/OFF		Open new log file or turn logging off
60	LR	r1	r2/i2	load register r1 from register r2/i2
61	LWL	r1		load register r1 with length of work area
62	MR	r1	r2/i2	multiply register r1 by register r2/i2
63	MVC	o1	l2	move to fs/ws field at offset from current ws rba
64	MVCF	o1	l2	move to field at offset o1 for length l2 from work area
65	MVCW	o1	l2	move from field at offset o1 for length l2 to work area
66	MVI	o1	i2	move immediate hex data i2 to file offset o1
67	NAME	file		Open data file. If no file is specified in GUI or command mode, the file selection dialog will be used. Find limit is reset to file length. This is identical to OPEN.
68	NOTEPAD			Start Notepad for use with log data pasted from clipboard. Use Notepad to edit or annotate the selected log data and then save, print, or email it as required.
69	NR	r1	r2/i2	and register r1 with r2/i2
70	OPEN	file		Open data file. If no file is specified in GUI or command mode, the file selection dialog will be used. Find limit is reset to file length.
71	OPENA	file		Open text file of type .DAT for input via ACCEPT
72	OR	r1	r2/i2	or register r1 with r2/i2
73	PASTE			Paste clipboard text to end of current GUI command line
74	PERM			Display current security manager permissions

## SuperZap User Guide V1.0.09 by Automated Software Tools

75	PSW			Display current program status information including registers, zap file instruction line, and condition code
76	RBA	o1		set current file offset to relative byte address o2. Value can be <hex> or <+hex> or <-hex>
77	REL			Current release level for OS, Java, and SuperZap
78	REP	o1	i2	Replace data at offset with specified data if no verify errors. This command is disabled on the Action menu if verify errors have occurred.
79	RESET			reset verify ok, reset error count, and reset program storage.
80	SAVE			Save any pending data file updates still in buffer. This command is disabled on the file menu unless there is an update pending. This command insures data file which may be accessed by other tasks is current. See demo2 for example usage of superzap to monitor online file updates.
81	SFK	o1	l2	set working storage offset and length of find key
82	SFL	l1		set file length and pad with zeros of longer
83	SH	r1	o2	subtract half word at file hex offset o2 from r1
84	SHOWT	o1	l2	Display all ASCII/EBCDIC printable text found
85	SHOWW	o1	l2	Display only ASCII/EBCDIC printable text combined in 3 to 80 character words. This removes most of the spurious text occurring in hex fields.
86	SIZE	x	y	Set width and height of GUI window in decimal pixels. Size limited to fit on screen.
87	SR	r1	r2/i2	subtract r2/i2 from r1
88	ST	r1	o2	store r1 in 4 bytes at file hex offset o2
89	STATUS	seconds		set default number of seconds for status updates on long running tasks such as FIND, ZAP commands, and Windows commands. The default is 5 seconds.
90	STC	r1	o2	store character from r1 to byte at file hex offset o2
91	STCK	o1	l2	store current date and time in format mm/dd/yy hh:mm:ss with length up to 17 characters
92	STCKF	o1	l2	store data and time of last file update in format mm/dd/yy hh:mm:ss with length up to 17 characters
93	STH	r1	o2	store half word at fs/ws offset
94	SUPPORT			open browser to Internet website support page

## SuperZap User Guide V1.0.09 by Automated Software Tools

95	SWL	r1		set working storage size and reset to zero.
96	TIMEOUT	seconds		set zap command timeout in seconds. Zero turns off timeout. The default is 3 seconds for CMD and ZAP commands.
97	TITLE	'text'		Set GUI window title
98	TM	o1	m2	test bits under mask and set Ones if all on, Equal if none on else Low
99	TMR	r1	m2	test bits under mask and set Ones if all on, Equal if none on else Low
100	TOOLBAR	ON/OFF		turn toolbar display on or off
101	TRACE	ALL/REGS/PSW/OFF		set trace to display current instruction with symbols resolved plus all the registers and PSW condition code and current fs/ws rba values.
102	USING	FS/WS		set current storage access mode to FS (file storage, default) or WS (working storage in memory)
103	VER	o1	i2	Verify data at specified offset
104	VIEWF			Set default dump view to complete file. This is default if file length < 1 million.
105	VIEWP			Set default dump view to single visible page. This is default if file size is over 1 million.
106	VIEWL			Set default dump view to single line
107	XLATE	o1	l2	translate fs/ws field or entire file to current ascii or ebcdic text
108	XR	r1	r2/i2	exclusive or register r1 with r2/i2
109	file	parms		execute zap file with SuperZap commands with parameters substituted in place of %1 - %9 occurring in zap macro command file model statements
110	*			comments. Note comments can also appear on commands separated from operands by spaces. To add a comment to a command with no operands, precede comments with comma and a space.

### SuperZap command operands:

file	ASCII file name space delimited. Must be in double quotes if long spacey name. May contain explicit path or may default to current directory set by CD.
o1	offset in fs or ws (see USING, DEC, HEX)
l1	length of field (see DEC, HEX)
x, y	window position or size in pixels
r1, r2	any of the 15 registers in the form r0 - rf

## SuperZap User Guide V1.0.09 by Automated Software Tools

i1, i2, m2          hex or text literal of any length (use single or double quotes for text)

SuperZap startup commands:

zap.bat	/G	options	start standalone GUI application as separate task
zap.bat	/C	options	
			start command line application
zap.bat	file	options	
			start batch zap command file

Notes:

1. Opcodes do not need to be preceded by spaces.
2. The second operand can be separated from opcode by spaces or comma.
3. Additional parms following second operand must be separated by commas.
4. First space after second operand delimits operands from following comments.
5. If opcode has no operands, then comments must be preceded by comma and space.
6. The second operand of register instructions can be register or hex immediate value.
7. Additional startup optional parms can be added as follows:
  - /RT - do not display timestamps to simplify regression test file compares
  - /NP - no permissions to test demo restrictions
  - /NT - no time stamp on default log file name
  - /SC - run following Windows bat command after switching to CMD mode
  - /SZ - run following startup zap command file to set user options such as
    1. Set window size and location with SIZE and LOC commands
    2. Set font size preference
    3. Open specific file and display information
    4. Run Windows commands to display information
    5. Run zap command program to display information

# SuperZap User Guide V1.0.09 by Automated Software Tools

## Appendix II: SuperZap Error Messages:

SuperZap Error Messages for Version 1.0 as of 01/04/05

Error ID	Description	Notes:
S001	IO open error on log file	Check if file in use by other application.
S002	IO write error on log file	Check if file in use by other application.
S003	IO close error on log file	Check if file in use by other application.
S004	IO open error on command file	Check if file in use by other application.
S005	IO read error on command file	Check if file in use by other application.
S006	IO close error on command file	Check if file in use by other application.
S007	IO open error on data file	Check if file in use by other application.
S008	IO read error on data file	Check if file in use by other application.
S009	IO write error on data file	Check if file in use by other application.
S010	IO close error on data file	Check if file in use by other application.
S011	Command limit exceeded	Abort if batch mode, else terminate current command file. Initial limit is 4096. See ACTR command to reset limit.
S012	Nested command file limit exceeded	Abort if batch mode, else terminate current command file. Limit is 20.
S013	Maximum errors exceeded	Abort if batch mode. Limit is 100
S015	Permission to write log file in current directory denied	See Java PolicyTool security page on <a href="http://www.SuperZap.net">www.SuperZap.net</a> for more information.
S016	Notepad.exe not found	Use CMD with correct path and program name for specific system
S017	Permission to execute command from Java denied	See Java PolicyTool to set permissions.
S018	Data file not open	Open file before issuing file commands
S020	Find key not defined	Use SFK to define key field in working storage or specific literal hex or text key as second parameter for FIND
S021	Missing offset parm	Specify hex offset or *+-hex offset
S022	Missing CD directory	CD select dialog not supported in batch
S023	Permission for CD directory change denied	See Java PolicyTool to set permissions.

## SuperZap User Guide V1.0.09 by Automated Software Tools

S024	COPYLOG command not available in command mode	Copying log to clipboard is only supported in GUI mode. Alternatively, use the LOG command to switch log files and then access the previous log file as a data file for SuperZap or other applications.
S025	missing open file name	Open select dialog not supported in batch. Specify file path and name to open.
S026	Permission to write or create file denied	See Java PolicyTool to set permissions.
S027	File not found	To create a new file, specify length as second hex operand
S028	File length override ignored for existing file or read only mode	See Java PolicyTool to set permissions. Use separate SFL command to change file length of existing file.
S029	Files over 2 GB not supported	Go to <a href="http://www.SuperZap.net">www.SuperZap.net</a> to download and install current version. Future release may be enhanced to support files larger than 2 GB.
S030	Use dummy demo 256 byte file containing hex 00 to FF since file read and write permission is denied.	See Java PolicyTool to set permissions.
S031	update suppressed due to error	Any verify error prevents following REP and other updates. To reset error flag, use RESET or OPEN file again.
S032	invalid REP parm	Hex offset and data value required.
S033	Replace command suppressed due to write permission denied	See Java PolicyTool to set permissions.
S034	Operation suppressed due to verify error	
S035	invalid VER parm	Hex offset and data value required.
S036	invalid hex or text parm value	Text must be in single or double quotes, hex must be 0-f with no quotes
S037	directory not found	Use CD with no operands to use change directory dialog in GUI or command mode to see all directories and select desired one

## SuperZap User Guide V1.0.09 by Automated Software Tools

S038	Permission for CD directory selection popup dialog denied	See Java PolicyTool to set permissions.
S039	Permission for file selection popup dialog denied	See Java PolicyTool to set permissions.
S040	Permissions denied due to user specified /NP parm for testing	See Java PolicyTool to set permissions.
S041	Start execution error occurred	Some Windows systems may not support start command for browser, notepad, or email. Start manually or with explicit CMD command and then paste clipboard to proceed.
S042	Branch label undefined	If using forward branch labels run as batch command which preloads source file to define all branch labels and then executes it interpretively with code optimizations.
S043	invalid register	must be Rn where n is 0-f indicating one of 16 registers.
S044	division by 0	result set to 1
S045	maximum label stack limit of 100 exceeded.	See RESET command.
S046	maximum command stack limit of 10000 exceeded	See RESET command.
S047	invalid digit or sign in packed field	dump or display field in hex to verify that last nibble in field is valid sign (hex C, D, or F).
S048	attempt to access beyond end of file	Verify length of file via LFL command
S049	invalid DISPLAY parm	First parm must be hex file offset if present else * assumed. Remaining parms must be 'text', Hn, In, Pn, Rn, or Tn separated by commas and no spaces. See demo5 for example.
S050	missing immediate data parm	Immediate hex and/or text data in the form of 'text' or "text" separated by commas. The maximum size of immediate data is 256.
S053	maximum nested commands exceeded	Limit set to 20.
S054	interactive command not supported in batch mode	COPYLOG, GUIDE, NOTEPAD, SUPPORT, and CMD interactive mode not supported in batch. Individual non-interactive CMD commands such as DIR or ERASE are supported.

## SuperZap User Guide V1.0.09 by Automated Software Tools

S055	invalid macro command parm reference	zap commands may contain %1 thru %9 parameter place holders for substitution of parm values at execution time. %% is replaced with % and all other %? Combinations cause error.
S056	SFL requested file length invalid	must be 0 to 2 GB
S057	IO error on set file length	Check if file in use by other application. or if sufficient disk space is available for the file size specified.
S058	IO error on accept file open	Verify file.dat exists and is not in use by other program.
S059	IO error on accept file read	Verify file.dat exists and is not in use by other program.
S060	accept is only supported when issued from zap file versus from command line.	Place zap commands including accept in a zap file and then execute the zap file from GUI or command line interface. Note zap commands run as a separate thread and wait for input at each accept. You can cancel a zap command at any time using CTRL-BREAK in GUI or CTRL-C in command mode.
S061	working storage offset invalid	default size of working storage is 4096. Use SWL set working storage length command to reset size up to limits of real storage.
S062	invalid hex length	hex length must be 1 to 256 for most instructions.
S063	font size outside New Courier fixed width font limits 8 - 72	Use FONT command to change default font size of 12. You can also hide the toolbar and status line to provide more space on screen for scrollable output log when using larger fonts for visibility.
S064	location for window is out of maximum window range	When resizing and moving location of window with SIZE and LOC commands, these commands will adjust size and location to keep minimum size screen of 150 by 150 visible.
S065	window resizing only supported in GUI mode	Graphical commands not supported in command mode or batch where the required objects are not available.

## SuperZap User Guide V1.0.09 by Automated Software Tools

S066	command execution interruption error	Startup of real time monitor, command processor, or zap command thread failed. Verify that required permissions such as thread modify are enabled via Policy manager.
S067	command execution output error	The Windows command processor has failed. Check for file conflicts and retry.
S068	command execution input error	The Windows command processor has failed. Check for file conflicts and retry.
S069	command execution time out error	The current timeout limit has been reached and command has been cancelled. Use TIMEOUT command to change timeout limit or turn off using 0.
S070	command execution cancelled	Command has been cancelled due to timeout, CTRL-BREAK, or CTRL-C.
S071	command execution ended with non zero return code	An error occurred in Windows command processor causing process to end with non-zero return code indicated
S072	command execution still running - cancel or retry command later	A Windows command started with CMD prefix is still running. Cancel it via CTRL-BREAK or CTRL-C in command mode. This condition will occur if Windows command is expecting input which requires that you switch to CMD mode and then enter the Windows command so requested input can be entered via command line.
S074	missing log file parm	The LOG command requires file name for new log or OFF to turn off logging.

## SuperZap User Guide V1.0.09 by Automated Software Tools

S075	invalid startup parm option	SuperZap startup requires java or javaw -classpath .;superzap.jar followed by batch zap file name or any of the following startup parms: /G - GUI mode, /C interactive command line mode, /RT - suppress all time stamps for regression test file comparison use, /NT suppress time stamp on log file names, /SC execute following Windows bat file after switching to CMD mode, /SZ execute following zap file at startup in GUI or command mode. See zap.bat command and demos for source code examples.
S076	wait command only allowed in zap command files	The WAIT command cannot be issued from GUI command line or interactive command line since both already have wait loop waiting for next input event.
S077	wait interrupted	An unexpected interruption has occurred during wait interval. Please report any unexplained errors along with zap file to recreate problem if possible.
S078	Aborting due to external shutdown request	A CTRL-BREAK or CTRL-C request has terminated process.
S079	Duplicate label ignored in zap command file	The first definition of a label starting with period is used and duplicates are ignored.
S080	Data file save update I/O error - file may be corrupted	An I/O error has occurred during data file update. This could occur due to file conflict or space problem. Verify file status and recreate if necessary.
S081	Data file length not equal specified length	If the size parameter is included on OPEN or NAME command, and the length of existing file does not match, this error will be issued. To change size of existing file after open, use SFL command.
S082	Maximum symbol EQU definitions exceeded	Limit is 100 per zap file
S083	Conflicting WS: and FS: offset overrides	Correct expression to only include one.
S084	Invalid TRACE option ALL, REGS, PSW, OFF	Correct parm to specify valid option
S085	invalid command code	Command codes must start with character >= 'A'

# **SuperZap User Guide V1.0.09 by Automated Software Tools**

# SuperZap User Guide V1.0.09 by Automated Software Tools

## Appendix III: Examples:

1. **demo1.zap** - dump file in hex, ASCII, and EBCDIC
2. **demo2.zap** – find, verify, and replace
3. **demo3.zap** - create a test file using macro instruction loop
4. **demo4.zap** – create a test file in ASCII and EBCDIC and find record
5. **demo5.zap** – create a test file and formatted report with field totals
6. **demo6.zap** – generate a file containing prime numbers
7. **demo7.zap** – monitor status of real time file updates in a separate process
8. **demo8.zap** – resize scrollable GUI window and font size for usability
9. **demo9.zap** – show benchmark statistics using status line display
10. **demo10.zap** – show interactive file query running from GUI, command line, or Windows interactive command mode

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo1.zap - dump file in hex, ASCII, and EBCDIC

Source command file:

```
* demo1.zap - dump file in hex, ascii, and ebcdic
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo opens data file containing the 256 bytes from hex 00 to hex
ff
* and displays the contents first in default ascii mode and then again
* in ebcdic character set mode
*
title 'demo1 - open demo.dat and display in hex and ASCII and EBCDIC
text'
name demo.dat
dump
dumph
dumplt
showt
ebcdic
dump
dumplt
showt
```

Generated demo1.log file using command zap /g /sz demo1

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:07:06
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070706
.log
Enter command or help
* demo1.zap - dump file in hex, ascii, and ebcdic
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo opens data file containing the 256 bytes from hex 00 to hex
ff
* and displays the contents first in default ascii mode and then again
* in ebcdic character set mode
*
title 'demo1 - open demo.dat and display in hex and ASCII and EBCDIC
text'
name demo.dat
Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo.dat
Data file Date = 01/06/05 Time = 07:36:43
Data file len(hex) = 100 len(dec) = 256
dump
00000000 *00010203 04050607 08090A0B 0C0D0E0F* *.....*
00000010 *10111213 14151617 18191A1B 1C1D1E1F* *.....*
00000020 *20212223 24252627 28292A2B 2C2D2E2F* * !"#$%&'()*+,-./
00000030 *30313233 34353637 38393A3B 3C3D3E3F* *0123456789:;<=>?*
00000040 *40414243 44454647 48494A4B 4C4D4E4F* *@ABCDEFGHIJKLMNO*
00000050 *50515253 54555657 58595A5B 5C5D5E5F* *PQRSTUVWXYZ[\]^_
00000060 *60616263 64656667 68696A6B 6C6D6E6F* *`abcdefgijklmno*
```

Copyright 2005 Automated Software Tools Corporation.  
This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
00000070 *70717273 74757677 78797A7B 7C7D7E7F* *pqrstuvwxyz{|}~.*
00000080 *80818283 84858687 88898A8B 8C8D8E8F* *.....*
00000090 *90919293 94959697 98999A9B 9C9D9E9F* *.....*
000000A0 *A0A1A2A3 A4A5A6A7 A8A9AAAB ACADAEAF* *.....*
000000B0 *B0B1B2B3 B4B5B6B7 B8B9BABB BCBDBEBF* *.....*
000000C0 *C0C1C2C3 C4C5C6C7 C8C9CACB CCCDCECF* *.....*
000000D0 *D0D1D2D3 D4D5D6D7 D8D9DADB DCDDDEDF* *.....*
000000E0 *E0E1E2E3 E4E5E6E7 E8E9EAEB ECEDEEEF* *.....*
000000F0 *F0F1F2F3 F4F5F6F7 F8F9FAFB FCFDFEFF* *.....*
dumph
00000000 *00010203 04050607 08090A0B 0C0D0E0F 10111213 14151617
18191A1B 1C1D1E1F*
00000020 *20212223 24252627 28292A2B 2C2D2E2F 30313233 34353637
38393A3B 3C3D3E3F*
00000040 *40414243 44454647 48494A4B 4C4D4E4F 50515253 54555657
58595A5B 5C5D5E5F*
00000060 *60616263 64656667 68696A6B 6C6D6E6F 70717273 74757677
78797A7B 7C7D7E7F*
00000080 *80818283 84858687 88898A8B 8C8D8E8F 90919293 94959697
98999A9B 9C9D9E9F*
000000A0 *A0A1A2A3 A4A5A6A7 A8A9AAAB ACADAEAF B0B1B2B3 B4B5B6B7
B8B9BABB BCBDBEBF*
000000C0 *C0C1C2C3 C4C5C6C7 C8C9CACB CCCDCECF D0D1D2D3 D4D5D6D7
D8D9DADB DCDDDEDF*
000000E0 *E0E1E2E3 E4E5E6E7 E8E9EAEB ECEDEEEF F0F1F2F3 F4F5F6F7
F8F9FAFB FCFDFEFF*
dumpr
00000000 *..... !"#%&'()*+,-
./0123456789:;<=>?*
00000040
*@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxy z{|}~.*
00000080
*.....*
000000C0
*.....*
showt
!"#%&'()*+,-
./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr
stuvwxy z{|}~
Total ASCII text characters found = 95
ebcdic
dump
00000000 *00010203 04050607 08090A0B 0C0D0E0F* *.....*
00000010 *10111213 14151617 18191A1B 1C1D1E1F* *.....*
00000020 *20212223 24252627 28292A2B 2C2D2E2F* *.....*
00000030 *30313233 34353637 38393A3B 3C3D3E3F* *.....*
00000040 *40414243 44454647 48494A4B 4C4D4E4F* *.....<(+|*
00000050 *50515253 54555657 58595A5B 5C5D5E5F* *&.....!$*);*
00000060 *60616263 64656667 68696A6B 6C6D6E6F* *-/.....,%_>?*
00000070 *70717273 74757677 78797A7B 7C7D7E7F* *.....`:#@'="*
00000080 *80818283 84858687 88898A8B 8C8D8E8F* *.abcdefghi.....*
00000090 *90919293 94959697 98999A9B 9C9D9E9F* *.jklmnopqr.....*
000000A0 *A0A1A2A3 A4A5A6A7 A8A9AAAB ACADAEAF* *.~stuvwxy z.....*
000000B0 *B0B1B2B3 B4B5B6B7 B8B9BABB BCBDBEBF* *^.....[ ].....*
000000C0 *C0C1C2C3 C4C5C6C7 C8C9CACB CCCDCECF* *{ABCDEFGHI.....*
000000D0 *D0D1D2D3 D4D5D6D7 D8D9DADB DCDDDEDF* *}JKLMNOPQR.....*
000000E0 *E0E1E2E3 E4E5E6E7 E8E9EAEB ECEDEEEF* *\.STUVWXYZ.....*
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
000000F0 *F0F1F2F3 F4F5F6F7 F8F9FAFB FCFDFEFF* *0123456789.....*
dump
00000000
*.....*
00000040 * .....<(+|&.....!$*);.-
/.....,%_>?.....`:#@'="*
00000080
*.abcdefghi.....jklmnopqr.....~stuvwxyz.....^.....[].....*
000000C0
*{ABCDEFGHI.....}JKLMNOPQR.....\STUVWXYZ.....0123456789.....*
showt

.<(+|& !$*); -/ ,%_>? `:#@'=" abcdefghi jklmnopqr ~stuvwxyz ^ []
{ABCDEFGHI }JKL
MNOPQR \ STUVWXYZ 0123456789
Total EBCDIC text characters found = 95
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo2.zap - find, verify, and replace

Source command file demo2.zap:

```
* demo2.zap - find, verify, and replace data in file demo2.dat
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates demo2.dat and then finds 2 occurrences of hex 1234
* and replace each occurrence with 'AB' using default ascii character
set
*
  title 'demo2 - create demo2.dat and verify and replace data'
  cmd erase demo2.dat
  open demo2.dat 8  create new demo2.dat with 8 bytes of zeros
  ver  00 0000
  rep  00 1234
  ver  04 0000
  rep  04 1234
  dump
  rba  0
.loop
  find * 1234
  bnz  .eof
  ver  * 1234
  rep  * 'AB'
  b   .loop
.eof
  dump
```

Output log file from command zap /g /sz demo2

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:07:20
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070720
.log
Enter command or help
* demo2.zap - find, verify, and replace data in file demo2.dat
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates demo2.dat and then finds 2 occurrences of hex 1234
* and replace each occurrence with 'AB' using default ascii character
set
*
  title 'demo2 - create demo2.dat and verify and replace data'
  cmd erase demo2.dat
Starting Windows command process
  open demo2.dat 8  create new demo2.dat with 8 bytes of zeros
Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo2.dat
Data file len(hex) = 8  len(dec) = 8
  ver  00 0000
  rep  00 1234
  ver  04 0000
  rep  04 1234
```

**Copyright 2005 Automated Software Tools Corporation.**  
**This is part of SuperZap distributed under open source GPL License.**

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
dump
00000000 *12340000 12340000          * *.4...4..      *
  rba 0
RBA hex file offset = 0
.loop
  find * 1234
Find starting at offset = 00000000 key = 1234
00000000 *12340000 12340000          * *.4...4..      *
  bnz .eof
  ver * 1234
  rep * 'AB'
  b .loop
  find * 1234
Find starting at offset = 00000000 key = 1234
00000004 *12340000          * *.4..          *
  bnz .eof
  ver * 1234
  rep * 'AB'
  b .loop
  find * 1234
Find starting at offset = 00000004 key = 1234
Find key not found
  bnz .eof
dump
00000000 *41420000 41420000          * *AB..AB..      *
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo3.zap - create a test file using macro instruction loop

Source command file:

```
* demo3.zap - create test file demo3.dat using zap commands
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates a test file demo3.dat containing hex 00-FF
*
title 'demo3 - create demo3.dat with hex 00 to FF'
erase demo3.dat
name demo3.dat 100 create file with zeros
lr r0,0
echo off
.loop
stc r0,*
rba *+1
ar r0,1
cr r0,100
bl .loop
dump
```

Output log file demo3.log from command zap /g /sz demo3

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:07:39
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070739
.log
Enter command or help
* demo3.zap - create test file demo3.dat using zap commands
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates a test file demo3.dat containing hex 00-FF
*
title 'demo3 - create demo3.dat with hex 00 to FF'
erase demo3.dat
Starting Windows command process
name demo3.dat 100 create file with zeros
Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo3.dat
Data file len(hex) = 100 len(dec) = 256
lr r0,0
echo off
00000000 *00010203 04050607 08090A0B 0C0D0E0F* *.....*
00000010 *10111213 14151617 18191A1B 1C1D1E1F* *.....*
00000020 *20212223 24252627 28292A2B 2C2D2E2F* * !"#$%&'()*+,-./
00000030 *30313233 34353637 38393A3B 3C3D3E3F* *0123456789:;<=>?*
00000040 *40414243 44454647 48494A4B 4C4D4E4F* *@ABCDEFGHIJKLMNO*
00000050 *50515253 54555657 58595A5B 5C5D5E5F* *PQRSTUVWXYZ[\]^_*
00000060 *60616263 64656667 68696A6B 6C6D6E6F* *`abcdefghijklmno*
00000070 *70717273 74757677 78797A7B 7C7D7E7F* *pqrstuvwxyz{|}~.*
00000080 *80818283 84858687 88898A8B 8C8D8E8F* *.....*
00000090 *90919293 94959697 98999A9B 9C9D9E9F* *.....*
000000A0 *A0A1A2A3 A4A5A6A7 A8A9AAAB ACADAEAF* *.....*
```

Copyright 2005 Automated Software Tools Corporation.  
This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
000000B0 *B0B1B2B3 B4B5B6B7 B8B9BABB BCBDBEBF* *.....*
000000C0 *C0C1C2C3 C4C5C6C7 C8C9CACB CCCDCECF* *.....*
000000D0 *D0D1D2D3 D4D5D6D7 D8D9DADB DCDDDEDF* *.....*
000000E0 *E0E1E2E3 E4E5E6E7 E8E9EAEB ECEDEEEF* *.....*
000000F0 *F0F1F2F3 F4F5F6F7 F8F9FAFB FCFDFEFF* *.....*
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo4.zap - create a test file in ASCII and EBCDIC and find record

Source command file:

```
* demo4.zap - create test file in ascii and ebcdic and find record
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* create ascii 80 byte file from ascii text file
*
title 'demo4 - create demo4.dat in ASCII and EBCDIC and find record'
cmd erase demo4.dat
name demo4.dat 50      output output with room for 1 80 byte record
opena demo4.txt       open input ascii text file for accept
.loop1
accept *,50           read ascii record into current file area
bz .eof1              exit if end of input file
rba *+50              position to next record area in file
sfl *+50              set new file length + 50 hex
b .loop1              do it again
.eof1
sfl *                  truncate file length to include last record
dump , dump ascii 80 byte record file
find 0 'record 3'
close
*
* create ebcdic 80 byte file from ascii text file
*
ebcdic                switch to EBCDIC code set
cmd erase demo4.dat
name demo4.dat 50      output output with room for 1 80 byte record
opena demo4.txt       open input ascii text
.loop2
accept *,50           read ascii record into current file area as
EBCDIC
bz .eof2              exit if end of input file
rba *+50              position to next record area in file
sfl *+50              set new file length + 50 hex
b .loop2              do it again
.eof2
sfl *                  truncate file length to include last record
dump , dump ebcdic 80 byte record file
find 0 'record 3'
*
* translate ebcdic file to ascii file
*
ascii
xlate
dump
find 0 'record 3'
*
* define key field in working storage and use it to find record
*
using ws              switch to working storage
sfk 0,8              define 8 character key field for use by find
mvi 0,'record 3' initialize key field
*
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
* find record based on key in working storage key field
*
  find *
```

Output log from command zap /g /sz demo4

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:07:43
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070743
.log
Enter command or help
* demo4.zap - create test file in ascii and ebcdic and find record
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* create ascii 80 byte file from ascii text file
*
  title 'demo4 - create demo4.dat in ASCII and EBCDIC and find record'
  cmd erase demo4.dat
Starting Windows command process
  name demo4.dat 50      output output with room for 1 80 byte record
Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo4.dat
Data file len(hex) = 50 len(dec) = 80
  opena demo4.txt      open input ascii text file for accept
.loop1
  accept *,50          read ascii record into current file area
accept input record = record 1 demo4 test reocrd
abcdefghijklmnopqrstuvwxyz
  bz .eof1             exit if end of input file
  rba **+50            position to next record area in file
RBA hex file offset = 50
  sfl **+50           set new file length + 50 hex
  b .loop1            do it again
  accept *,50          read ascii record into current file area
accept input record = record 2 demo4 test record
ABCDEFGHIJKLMNQPQRSTUVWXYZ
  bz .eof1             exit if end of input file
  rba **+50            position to next record area in file
RBA hex file offset = A0
  sfl **+50           set new file length + 50 hex
  b .loop1            do it again
  accept *,50          read ascii record into current file area
accept input record = record 3 demo4 test record 0123456789
  bz .eof1             exit if end of input file
  rba **+50            position to next record area in file
RBA hex file offset = F0
  sfl **+50           set new file length + 50 hex
  b .loop1            do it again
  accept *,50          read ascii record into current file area
accept input record = null
  bz .eof1             exit if end of input file
  sfl *                truncate file length to include last record
  dump , dump ascii 80 byte record file
00000000 *7265636F 72642031 2064656D 6F342074* *record 1 demo4 t*
00000010 *65737420 72656F63 72642061 62636465* *est reocrd abcde*
```

Copyright 2005 Automated Software Tools Corporation.  
This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
00000020 *66676869 6A6B6C6D 6E6F7071 72737475* *fghijklmnopqrstu*
00000030 *76777879 7A202020 20202020 20202020* *vwxyz *
00000040 *20202020 20202020 20202020 20202020* * *
00000050 *7265636F 72642032 2064656D 6F342074* *record 2 demo4 t*
00000060 *65737420 7265636F 72642041 42434445* *est record ABCDE*
00000070 *46474849 4A4B4C4D 4E4F5051 52535455* *FGHIJKLMNOPQRSTU*
00000080 *56575859 5A202020 20202020 20202020* *VWXYZ *
00000090 *20202020 20202020 20202020 20202020* * *
000000A0 *7265636F 72642033 2064656D 6F342074* *record 3 demo4 t*
000000B0 *65737420 7265636F 72642030 31323334* *est record 01234*
000000C0 *35363738 39202020 20202020 20202020* *56789 *
000000D0 *20202020 20202020 20202020 20202020* * *
000000E0 *20202020 20202020 20202020 20202020* * *
    find 0 'record 3'
Find starting at offset = 00000000 key = 7265636F72642033
000000A0 *7265636F 72642033 2064656D 6F342074* *record 3 demo4 t*
    close
*
* create ebcdic 80 byte file from ascii text file
*
    ebcdic                switch to EBCDIC code set
    cmd erase demo4.dat
Starting Windows command process
    name demo4.dat 50      output output with room for 1 80 byte record
Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo4.dat
Data file len(hex) = 50 len(dec) = 80
    opena demo4.txt      open input ascii text
.loop2
    accept *,50          read ascii record into current file area as
EBCDIC
accept input record = record 1 demo4 test reocrd
abcdefghijklmnopqrstuvwxy
    bz .eof2             exit if end of input file
    rba *+50             position to next record area in file
RBA hex file offset = 50
    sfl *+50            set new file length + 50 hex
    b .loop2            do it again
    accept *,50          read ascii record into current file area as
EBCDIC
accept input record = record 2 demo4 test record
ABCDEFGHIJKLMNPOQRSTUVWXYZ
    bz .eof2             exit if end of input file
    rba *+50             position to next record area in file
RBA hex file offset = A0
    sfl *+50            set new file length + 50 hex
    b .loop2            do it again
    accept *,50          read ascii record into current file area as
EBCDIC
accept input record = record 3 demo4 test record 0123456789
    bz .eof2             exit if end of input file
    rba *+50             position to next record area in file
RBA hex file offset = F0
    sfl *+50            set new file length + 50 hex
    b .loop2            do it again
    accept *,50          read ascii record into current file area as
EBCDIC
accept input record = null
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

```
bz .eof2          exit if end of input file
sfl *            truncate file length to include last record
dump , dump ebcdic 80 byte record file
00000000 *99858396 998440F1 40848594 96F440A3* *record 1 demo4 t*
00000010 *85A2A340 99859683 99844081 82838485* *est reocrd abcde*
00000020 *86878889 91929394 95969798 99A2A3A4* *fghijklmnopqrstu*
00000030 *A5A6A7A8 A9404040 40404040 40404040* *vwxyz *
00000040 *40404040 40404040 40404040 40404040* * *
00000050 *99858396 998440F2 40848594 96F440A3* *record 2 demo4 t*
00000060 *85A2A340 99858396 998440C1 C2C3C4C5* *est record ABCDE*
00000070 *C6C7C8C9 D1D2D3D4 D5D6D7D8 D9E2E3E4* *FGHIJKLMNOPQRSTU*
00000080 *E5E6E7E8 E9404040 40404040 40404040* *VWXYZ *
00000090 *40404040 40404040 40404040 40404040* * *
000000A0 *99858396 998440F3 40848594 96F440A3* *record 3 demo4 t*
000000B0 *85A2A340 99858396 998440F0 F1F2F3F4* *est record 01234*
000000C0 *F5F6F7F8 F9404040 40404040 40404040* *56789 *
000000D0 *40404040 40404040 40404040 40404040* * *
000000E0 *40404040 40404040 40404040 40404040* * *
find 0 'record 3'
Find starting at offset = 00000000 key = 99858396998440F3
000000A0 *99858396 998440F3 40848594 96F440A3* *record 3 demo4 t*
*
* translate ebcdic file to ascii file
*
ascii
xlate
dump
00000000 *7265636F 72642031 2064656D 6F342074* *record 1 demo4 t*
00000010 *65737420 72656F63 72642061 62636465* *est reocrd abcde*
00000020 *66676869 6A6B6C6D 6E6F7071 72737475* *fghijklmnopqrstu*
00000030 *76777879 7A202020 20202020 20202020* *vwxyz *
00000040 *20202020 20202020 20202020 20202020* * *
00000050 *7265636F 72642032 2064656D 6F342074* *record 2 demo4 t*
00000060 *65737420 7265636F 72642041 42434445* *est record ABCDE*
00000070 *46474849 4A4B4C4D 4E4F5051 52535455* *FGHIJKLMNOPQRSTU*
00000080 *56575859 5A202020 20202020 20202020* *VWXYZ *
00000090 *20202020 20202020 20202020 20202020* * *
000000A0 *7265636F 72642033 2064656D 6F342074* *record 3 demo4 t*
000000B0 *65737420 7265636F 72642030 31323334* *est record 01234*
000000C0 *35363738 39202020 20202020 20202020* *56789 *
000000D0 *20202020 20202020 20202020 20202020* * *
000000E0 *20202020 20202020 20202020 20202020* * *
find 0 'record 3'
Find starting at offset = 00000000 key = 7265636F72642033
000000A0 *7265636F 72642033 2064656D 6F342074* *record 3 demo4 t*
*
* define key field in working storage and use it to find record
*
using ws          switch to working storage
sfk 0,8          define 8 character key field for use by find
mvi 0,'record 3' initialize key field
*
* find record based on key in working storage key field
*
find *
Find starting at offset = 00000000 key = 7265636F72642033
000000A0 *7265636F 72642033 2064656D 6F342074* *record 3 demo4 t*
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

SuperZap total errors = 0

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo5.zap - create test file and formatted report with field totals

Source command file for demo5.zap:

```
* demo5.zap - create test file and produce formatted report from file
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
      title 'demo5 - create demo5.dat and format report'
      echo off          turn off instruction listing on log
      dec              use decimal offsets and integer constants vs
hex
*
* file record layout
*
record equ fs:*          define record as current location in file
storage
key     equ record+0    10 character key
count  equ record+10   2 byte integer count
amount equ record+12   3 byte paced total
lrecl  equ 15
*
* build a test file with 3 records using subroutine demo5add.zap
*
cmd erase demo5.dat
open demo5.dat 45
demo5add 'charlie d' 10 -5
demo5add 'faith h' 20 15
demo5add 'shania t' 30 25
dump
close
*
* open test file and produce report with totals
*
* working storage layout
*
work          equ r0      work register to cvt between formats
total_count  equ r1      total of counts
total_amount equ r2      total of amounts
cur_record   equ r3      current record position in file
file_length  equ r4      length of file
total_records equ r5     total records processed
*
* read file and produce detail report with totals
*
console on echo all log output in batch to console
lr total_count,0
lr total_amount,0
lr total_records,0
open demo5.dat
lfl file_length
lr cur_record,0
.loop
cr cur_record,file_length are we at end of file
bnl .eof                  yes, go print totals
rba cur_record            position to current record in file
lh work,count
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

```
    ar  total_count,work          total count field
    field 3
    cvb work,amount
    ar  total_amount,work        total packed amount field
    display record,'record key = ',t10,'          count = ',i2,'          amount
= ',p3
    ar  total_records,1
    ar  cur_record,lrecl
    b   .loop
.eof
    display ' total records = ',total_records,' total count =
',total_count,' total amount = ',total_amount
Source command file for demo5all.zap subroutine called by demo5.zap:
```

```
* demo5add.zap - subroutine to add record to file
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* subroutine to add a record to current demo5 file
* parms are key, count, and amount
*
work1  equ r0
work2  equ r1
record equ fs:*          define record as current location in file
storage
key    equ record+0  10 character key
count  equ record+10  2 byte integer count
amount equ record+12  3 byte paced total
lrecl  equ 15
*
* add record at current file position and move to next record
*
    mvi key,'          ' init key to spaces
    mvi key,%1        set 10 char key field from literal
    lr  work1,%2
    sth work1,count   set half word count field from integer count
    lr  work1,%3
    field 3          set length of cvd/cvb packed fields (default 8)
    cvd work1,amount set 3 byte packed decimal amount from integer
amount
    rba record+lrecl  position to next record
```

Sample output log demo5.log:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:07:58
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070758
.log
Enter command or help
* demo5.zap - create test file and produce formatted report from file
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
    title 'demo5 - create demo5.dat and format report'
    echo off          turn off instruction listing on log
```

**Copyright 2005 Automated Software Tools Corporation.**  
**This is part of SuperZap distributed under open source GPL License.**

## SuperZap User Guide V1.0.09 by Automated Software Tools

Starting Windows command process

```
00000000 *63686172 6C696520 64200A00 00005D66* *charlie d ....]f*
00000010 *61697468 20682020 20140000 015C7368* *aith h ....\sh*
00000020 *616E6961 20742020 1E000002 5C      * *ania t ....\  *
record key = charlie d      count = 10      amount = -5
record key = faith h      count = 20      amount = 15
record key = shania t      count = 30      amount = 25
total records = 3 total count = 60 total amount = 35
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo6.zap - generate a file containing prime numbers

Source command file:

```
* demo6.zap - generate file demo6.dat containing prime numbers
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates file demo6.dat containing prime numbers in big
endian
* integer full word format. The file grows as each prime is added.
* See demo7.zap for separate process that monitors this file in window.
*
title 'demo6 - generate demo6.dat file with primes'
echo off                suppress commands on log
toolbar off             hide the toolbar to save window space
loc 0,250               move window below monitor.zap window
size 600,250           set window size to 600 x 250
timeout 0               turnoff timeout
console on              log to console if running in command mode
cmd erase demo6.dat
open demo6.dat 4        open existing file or create new file
big                     switch from little endian default to big endian
lr r1,3                 r1 = last prime number tested
st r1,0                 store first prime in file
lr r2,0                 r2 = offset to last prime added
display 'demo6.zap generates file demo6.dat with prime numbers'
.next_test_prime test next prime
ar r1,2                 r1 = next prime to test
lr r0,0                 r0 = offset of first prime divisor for odd
primes
.next_divisor          try dividing by previous primes found
actr 100               reset instruction limit for next divisor loop
rba r0                 reset to first prime in file
l r5,*                 r5 = next prime divisor
cr r5,0                is prime equal 0
be .error              error if divisor zero
lr r6,r1               r6 = trial prime
dr r6,r5               r6 = quotient
mr r6,r5               r6 = divisor * quotient
cr r6,r1               compare product with test prime
be .next_test_prime   if equal, then go to next test prime
bh .prime              if high, then we have next prime
ar r0,4                r0 = offset to next prime divisor
cr r0,r2               is r0 past last prime divisor
bnh .next_divisor     repeat test for all previous prime divisors
.prime
lr r2,r0               update r2 to new last divisor
rba r2                 position to new last prime
sfl r2+4               set file length + 4
st r1,*                store new prime in file of primes found
save                   force file updates for use by monitor now
display *,i4           show 4 byte integer at current file loc
b .next_test_prime    repeat until CTRL-BREAK or CTRL-C cancel
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

# SuperZap User Guide V1.0.09 by Automated Software Tools

Output superzap.log file from command zap /G /SZ demo6

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:04
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070804
.log
Enter command or help
* demo6.zap - generate file demo6.dat containing prime numbers
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo creates file demo6.dat containing prime numbers in big
endian
* integer full word format. The file grows as each prime is added.
* See demo7.zap for separate process that monitors this file in window.
*
  title 'demo6 - generate demo6.dat file with primes'
  echo off          supress commands on log
Timeout monitor for CMD and ZAP turned off
Starting Windows command process
demo6.zap generates file demo6.dat with prime numbers
  5
  7
  11
  13
  17
  19
  23
  29
  31
  37
  41
  43
  47
  53
  59
  61
  67
  71
  73
  79
  83
  89
  ...

  521
  523
  541
  547
  557
  563
  569
SuperZap total errors = 0
```

Dump of demo6.dat after exiting from GUI execution window:

**Copyright 2005 Automated Software Tools Corporation.**  
**This is part of SuperZap distributed under open source GPL License.**

# SuperZap User Guide V1.0.09 by Automated Software Tools

SuperZap V1.0.05

Copyright 2005 Automated Software Tools Corporation

Current Date = 01/10/05 Time = 07:58:48

Log file = D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap.log

Enter command or help

open demo6.dat

Data file D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo6.dat

Data file Date = 01/10/05 Time = 07:08:12

Data file len(hex) = 19C len(dec) = 412

dump

```
00000000 *00000003 00000005 00000007 0000000B* *.....*
00000010 *0000000D 00000011 00000013 00000017* *.....*
00000020 *0000001D 0000001F 00000025 00000029* *.....%...)*
00000030 *0000002B 0000002F 00000035 0000003B* *...+.../...5...;*
00000040 *0000003D 00000043 00000047 00000049* *...=...C...G...I*
00000050 *0000004F 00000053 00000059 00000061* *...O...S...Y...a*
00000060 *00000065 00000067 0000006B 0000006D* *...e...g...k...m*
00000070 *00000071 0000007F 00000083 00000089* *...q.....*
00000080 *0000008B 00000095 00000097 0000009D* *.....*
00000090 *000000A3 000000A7 000000AD 000000B3* *.....*
000000A0 *000000B5 000000BF 000000C1 000000C5* *.....*
000000B0 *000000C7 000000D3 000000DF 000000E3* *.....*
000000C0 *000000E5 000000E9 000000EF 000000F1* *.....*
000000D0 *000000FB 00000101 00000107 0000010D* *.....*
000000E0 *0000010F 00000115 00000119 0000011B* *.....*
000000F0 *00000125 00000133 00000137 00000139* *...%...3...7...9*
00000100 *0000013D 0000014B 00000151 0000015B* *...=...K...Q...[*
00000110 *0000015D 00000161 00000167 0000016F* *...]...a...g...o*
00000120 *00000175 0000017B 0000017F 00000185* *...u...{.....*
00000130 *0000018D 00000191 00000199 000001A3* *.....*
00000140 *000001A5 000001AF 000001B1 000001B7* *.....*
00000150 *000001BB 000001C1 000001C9 000001CD* *.....*
00000160 *000001CF 000001D3 000001DF 000001E7* *.....*
00000170 *000001EB 000001F3 000001F7 000001FD* *.....*
00000180 *00000209 0000020B 0000021D 00000223* *.....#*
00000190 *0000022D 00000233 00000239 * *...-...3...9 *
SuperZap total errors = 0
```

Copyright 2005 Automated Software Tools Corporation.

This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

demo7.zap - monitor status of real time file updates in GUI window

Source command file for demo7.zap:

```
* demo7.zap - monitor status of real time file updates in gui window
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* demo monitoring of demo6.dat file being generated by demo6.zap
* start the demo6 process to generate demo6.dat file with primes
  title 'demo7 - monitor demo6.dat prime file updates'
  cmd zap /g /sz demo6
  echo off          don't show each instruction on log
  toolbar off      don't show toolbar
  console on       display output on console if running in command mode
  timeout 0        cancel timeouts
  loc 0 0          set gui screen location
  size 600 250    set gui screen size
  big
  actr 1000000     reset max instruction count
  wait 1           wait until demo6 separate process is up and running
.loop
  open demo6.dat,INPUT open data file being monitored for input
  lfl r1           get file length in r1
  cr r1,r2         is file length the same as last time
  be .wait        yes, go wait again
  cr r1,8          does file have at least 2 prime slots yet
  bnh .wait       no, wait again
  lr r2,r1         no, save new file length
  using ws        switch to working storage
  stck 0,         store current data and time in work area
  rba 0           position to offset 0 in ws
  display 'Current date and Time = ',t17
  stckf 0,        store file last modified date in work area
  rba 0           position to offset 0 in ws
  display 'demo6.dat last update = ',t17
  display 'File length = ',r1
  using fs        return to file storage
  rba r1          position to end of file
.getprime
  l r0,*-8        load next to last prime saved by demo6.zap via save
  cr r0,0         verify prime is not zero
  bz .error
  display 'Last prime in demo6.dat = ',r0
.wait
  wait 2          wait 2 seconds
  b .loop         repeat monitoring process until cancelled, timeout,
actr
.error
  display 'monitor error reading next to last prime'
```

Source for demo7.bat command to launch GUI windows for demo6 and demo7:

```
zap /g /sz demo7
```

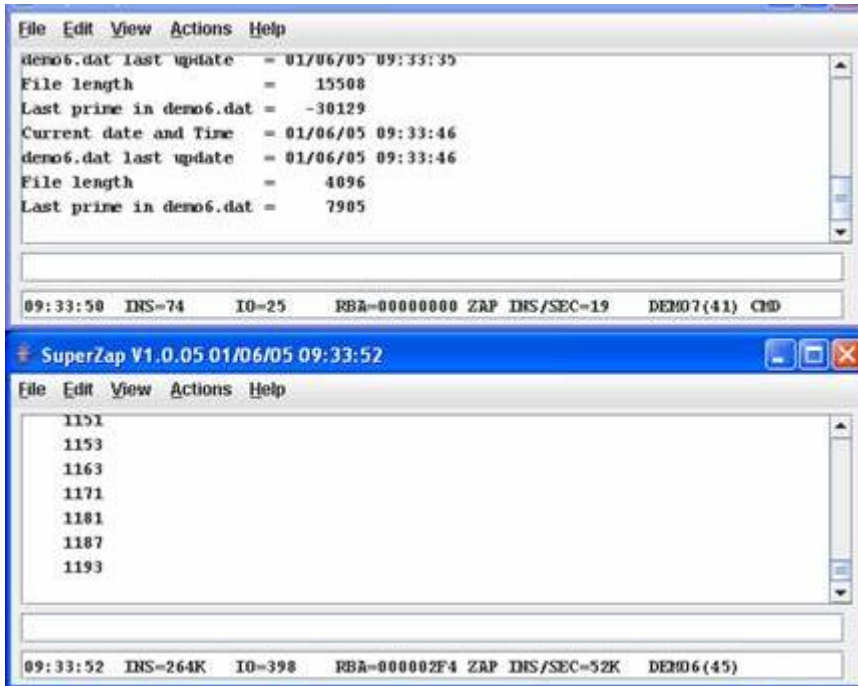
# SuperZap User Guide V1.0.09 by Automated Software Tools

Sample log for demo7:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:08
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070808
.log
Enter command or help
* demo7.zap - monitor status of real time file updates in gui window
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* demo monitoring of demo6.dat file being generated by demo6.zap
* start the demo6 process to generate demo6.dat file with primes
  title 'demo7 - monitor demo6.dat prime file updates'
  cmd zap /g /sz demo6
Starting Windows command process
D:\Work\eclipse\WORKSP~1\superzap\build\demos>java -classpath
"c:\program files\Automated Software Tools\SuperZap";"c:\program
files\Automated Software Tools\SuperZap\superzap.jar" superzap /g /sz
demo6
  echo off          don't show each instruction on log
Timeout monitor for CMD and ZAP turned off
Current date and Time    = 01/10/05 07:08:09
demo6.dat last update   = 01/10/05 07:08:06
File length              =          412
Last prime in demo6.dat =          563
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

Screen capture resulting from running demo7.bat:



# SuperZap User Guide V1.0.09 by Automated Software Tools

demo8.zap - resize scrollable GUI window and font size for usability

Source for demo8.zap:

```
* demo8.zap - resize scrollable GUI window for usability
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* resize gui window to 600 by 800 with 24 point font for visibility
* and then demo using Windows command mode to issue dir
*
  title 'demo8 - resize window and set large font'
  loc 0,0      position window to upper left
  size 800,600 set size to 800 by 600 pixils
  toolbar off  hide toolbar
  font 24      change to 24 point font for visibility
*
* issue dir to illustrate scrolling large font log
*
cmd dir
```

Sample log for demo8:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:16
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070816
.log
Enter command or help
* demo8.zap - resize scrollable GUI window for usability
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* resize gui window to 600 by 800 with 24 point font for visibility
* and then demo using Windows command mode to issue dir
*
  title 'demo8 - resize window and set large font'
  loc 0,0      position window to upper left
  size 800,600 set size to 800 by 600 pixils
  toolbar off  hide toolbar
  font 24      change to 24 point font for visibility
*
* issue dir to illustrate scrolling large font log
*
cmd dir
Starting Windows command process
Volume in drive D is DSHDAT
Volume Serial Number is D42C-BFE1
Directory of D:\Work\eclipse\WORKSP~1\superzap\build\demos
01/10/2005  07:08 AM    <DIR>          .
01/10/2005  07:08 AM    <DIR>          ..
01/06/2005  07:36 AM                256 demo.dat
10/16/2004  06:32 AM                10 demo1.dat
01/07/2005  09:36 AM                498 DEMO1.ZAP
01/07/2005  08:56 PM                905 DEMO10.BAT
01/07/2005  08:38 PM               1,238 DEMO10.ZAP
```

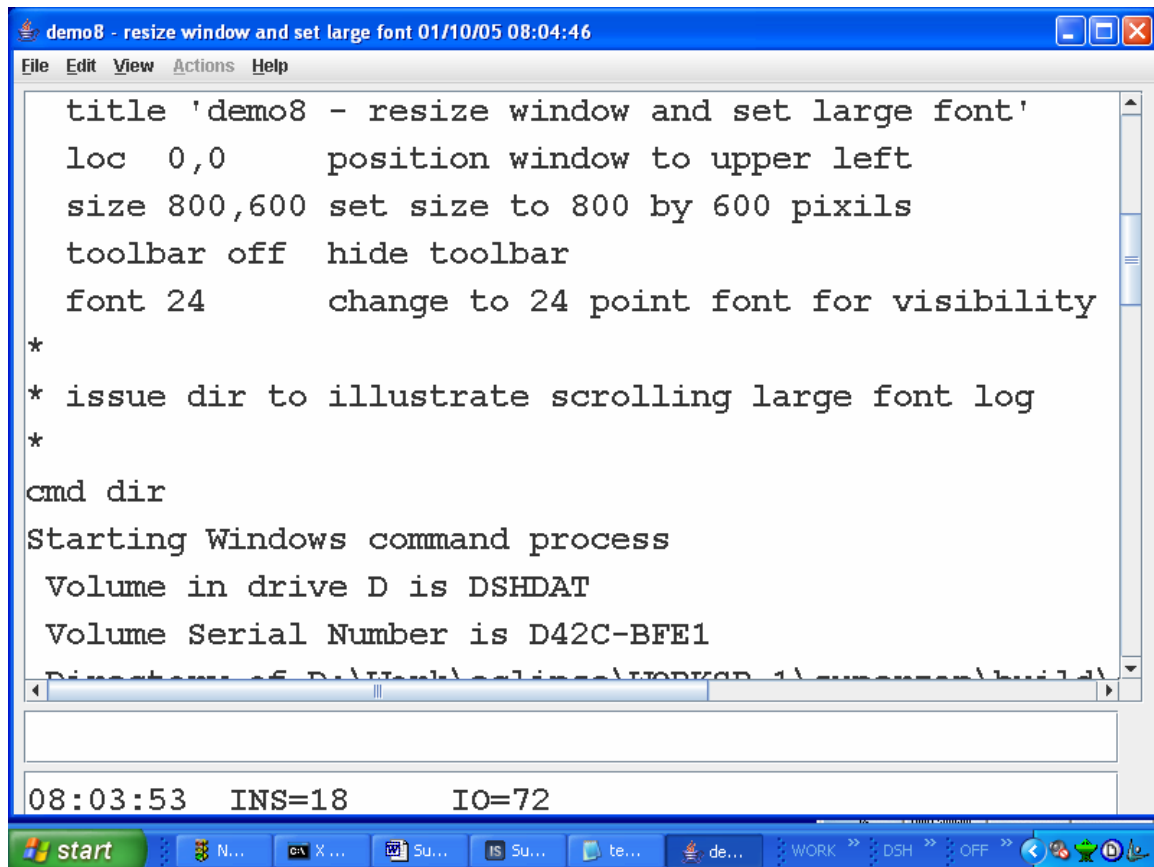
Copyright 2005 Automated Software Tools Corporation.  
This is part of SuperZap distributed under open source GPL License.

## SuperZap User Guide V1.0.09 by Automated Software Tools

```
01/10/2005 07:07 AM          8 demo2.dat
01/07/2005 09:36 AM        663 DEMO2.ZAP
01/10/2005 07:07 AM        256 demo3.dat
01/07/2005 09:36 AM        440 DEMO3.ZAP
01/10/2005 07:07 AM        320 demo4.dat
12/31/2004 11:00 AM        149 DEMO4.TXT
01/07/2005 09:36 AM     1,974 DEMO4.ZAP
01/10/2005 07:07 AM         45 demo5.dat
01/07/2005 09:36 AM     2,060 DEMO5.ZAP
01/06/2005 09:46 AM        969 DEMO5ADD.ZAP
01/10/2005 07:08 AM        412 demo6.dat
01/07/2005 09:36 AM     2,591 DEMO6.ZAP
01/07/2005 05:56 PM     1,975 DEMO7.ZAP
01/07/2005 09:36 AM        594 DEMO8.ZAP
01/07/2005 09:36 AM        785 DEMO9.ZAP
01/10/2005 06:58 AM        975 DEMOS.BAT
01/10/2005 07:07 AM     4,698 superzap_2005_0110_070706.log
01/10/2005 07:07 AM     1,672 superzap_2005_0110_070720.log
01/10/2005 07:07 AM     1,904 superzap_2005_0110_070739.log
01/10/2005 07:07 AM     8,078 superzap_2005_0110_070743.log
01/10/2005 07:08 AM     1,095 superzap_2005_0110_070758.log
01/10/2005 07:08 AM     1,968 superzap_2005_0110_070804.log
01/10/2005 07:08 AM     1,213 superzap_2005_0110_070808.log
01/10/2005 07:08 AM     2,218 superzap_2005_0110_070809.log
01/10/2005 07:08 AM          0 superzap_2005_0110_070816.log
01/10/2005 07:03 AM        172 ZAP.BAT
          31 File(s)        40,141 bytes
          2 Dir(s) 55,564,255,232 bytes free
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

Screen capture after starting demo8 using zap /g /sz demo8



```
demo8 - resize window and set large font 01/10/05 08:04:46
File Edit View Actions Help
title 'demo8 - resize window and set large font'
loc 0,0      position window to upper left
size 800,600 set size to 800 by 600 pixils
toolbar off  hide toolbar
font 24      change to 24 point font for visibility
*
* issue dir to illustrate scrolling large font log
*
cmd dir
Starting Windows command process
Volume in drive D is DSHDAT
Volume Serial Number is D42C-BFE1
Directory of D:\Hunk\colinas\WORKSP_1\superzap\build\
08:03:53  INS=18      IO=72
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

demo9.zap - use status statistics for benchmark testing

Source command file for demo9.zap:

```
* demo9.zap - show pc processor benchmark statistics using status line
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* this command will display instruction rate at 1 second intervals
* until 2 million branches have been executed. On a 3 GHZ Pentium IV
* processor using J2SE 1.4.2, this demo shows execution rate of about
* 500,000 branch instructions per second or 0.5 mips.
*
  title 'demo9 - branch loop benchmark'
  echo off
  dec          demimal values
  timeout 0    turn off timeout
  actr 2000000 set actr instruction limit to 2 million
  status 1     show status with instruction rate at 1 second
intervals
  console on   display status on console if running in batch mode
.loop
  b .loop
```

Output log for demo8.log:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:20
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070820
.log
Enter command or help
* demo9.zap - show pc processor benchmark statistics using status line
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* this command will display instruction rate at 1 second intervals
* until 2 million branches have been executed. On a 3 GHZ Pentium IV
* processor using J2SE 1.4.2, this demo shows execution rate of about
* 500,000 branch instructions per second or 0.5 mips.
*
  title 'demo9 - branch loop benchmark'
  echo off
Timeout monitor for CMD and ZAP turned off
Status logging interval set to 1 seconds
07:08:21  INS=499K   IO=19      ZAP INS/SEC=399K  DEMO9(18)
07:08:22  INS=1109K  IO=20      ZAP INS/SEC=488K  DEMO9(18)
07:08:23  INS=1712K  IO=21      ZAP INS/SEC=482K  DEMO9(18)
07:08:24  INS=2000K  IO=22      ZAP INS/SEC=484K  DEMO9(18)
SuperZap Error S011 current ACTR instruction limit exceeded
SuperZap total errors = 1
```

## SuperZap User Guide V1.0.09 by Automated Software Tools

Demo10.zap - show interactive file query application running in GUI, command line, or Windows interactive command line mode

Source for demo10.zap command file:

```
* demo10.zap - show interactive command line file query application
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo can be run in the following 4 modes:
* 1. Run as gui application          - zap /g /sz demo10.zap
* 2. Run as command line appl.      - zap /c /sz demo10.zap
* 3. Run as batch application        - zap demo10.zap
* 4. Run as gui Windows CMD appl.   - zap /g /sc demo10.bat
*
  title 'demo10 - interactive query of file demo.dat'
  echo off          turn off instruction logging
  console on       copy any output to console if batch mode
  timeout 0        turn off default 3 second timeout for zap command
  open demo.dat
key equ ws:0      define 1 byte search key in working storage
  sfk key,1       set default search key for find command
.loop
  display 'enter character to search for or q to exit'
  accept key,1    get search key from command line
  cli key,'q'    is this q for quit
  be .eof        yes, exit now
  display key,'search key = ',h1 no, display key in hex
  find 0
  b .loop        repeat until q or cancel
.eof
  display 'exiting now'
```

Source for Windows command startup file demo10.bat:

```
prompt $N$G
rem this demo10.bat Windows command file is executed as a interactive
Windows
rem command file from SuperZap GUI front end using the following
rem command to startup the GUI and the interactive command running
rem as a separate Windows process: zap /G /SC demo10.bat
rem
rem Note demo10.zap is run as an interactive batch application in
rem the Windows command process so when q is typed to terminate
rem the demo10.zap thread, the java batch process will terminate
rem and when the GUI frontend is terminated, the cmd process will
rem also terminate cleanly.
rem
rem this java interactive Windows application can be replaced with any
rem existing interactive Windows command line application thus using
rem SuperZap GUI as a front end to provide GUI features such as
rem scrollable output window and options to resize window and set font,
etc.
rem
prompt $P$G
zap demo10
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

Sample log output from running in gui mode using Windows CMD mode:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:56
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\demo10_2005_0110_070856.1
og
demo10
* demo10.zap - show interactive command line file query application
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo can be run in the following 4 modes:
* 1. Run as gui application      - zap /g /sz demo10.zap
* 2. Run as command line appl.  - zap /c /sz demo10.zap
* 3. Run as batch application    - zap demo10.zap
* 4. Run as gui Windows CMD appl. - zap /g /sc demo10.bat
*
  title 'demo10 - interactive query of file demo.dat'
  echo off          turn off instruction logging
Timeout monitor for CMD and ZAP turned off
enter character to search for or q to exit
search key = 61
00000061 *61626364 65666768 696A6B6C 6D6E6F70* *abcdefghijklmnop*
enter character to search for or q to exit
search key = 78
00000078 *78797A7B 7C7D7E7F 80818283 84858687* *xyz{|}~.....*
enter character to search for or q to exit
exiting now
SuperZap total errors = 0
```

Sample log from running demos.bat from GUI in CMD mode:

```
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:28
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070828
.log
Enter command or help
* demo10.zap - show interactive command line file query application
* copyright 2005 Automated Software Tools Corporation
* visit www.SuperZap.net for more information
*
* This demo can be run in the following 4 modes:
* 1. Run as gui application      - zap /g /sz demo10.zap
* 2. Run as command line appl.  - zap /c /sz demo10.zap
* 3. Run as batch application    - zap demo10.zap
* 4. Run as gui Windows CMD appl. - zap /g /sc demo10.bat
*
  title 'demo10 - interactive query of file demo.dat'
  echo off          turn off instruction logging
Timeout monitor for CMD and ZAP turned off
enter character to search for or q to exit
search key = 61
00000061 *61626364 65666768 696A6B6C 6D6E6F70* *abcdefghijklmnop*
```

**Copyright 2005 Automated Software Tools Corporation.**  
**This is part of SuperZap distributed under open source GPL License.**

## SuperZap User Guide V1.0.09 by Automated Software Tools

```
enter character to search for or q to exit
search key = 78
00000078 *78797A7B 7C7D7E7F 80818283 84858687* *xyz{|}~.....*
enter character to search for or q to exit
exiting now
SuperZap total errors = 0
SuperZap V1.0.05
Copyright 2005 Automated Software Tools Corporation
Current Date = 01/10/05 Time = 07:08:55
Log file =
D:\Work\eclipse\WORKSP~1\superzap\build\demos\superzap_2005_0110_070855
.log
Enter command or help
Starting Windows command process
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
D:\Work\eclipse\WORKSP~1\superzap\build\demos>demo10.bat
D:\Work\eclipse\WORKSP~1\superzap\build\demos>prompt $N$G
D>rem this demo10.bat Windows command file is executed as a interactive
Windows
D>rem command file from SuperZap GUI front end using the following
D>rem command to startup the GUI and the interactive command running
D>rem as a separate Windows process: zap /G /SC demo10.bat
D>rem
D>rem Note demo10.zap is run as an interactive batch application in
D>rem the Windows command process so when q is typed to terminate
D>rem the demo10.zap thread, the java batch process will terminate
D>rem and when the GUI front end is terminated, the cmd process will
D>rem also terminate cleanly.
D>rem
D>rem this java interactive Windows application can be replaced with
any
D>rem existing interactive Windows command line application thus using
D>rem SuperZap GUI as a front end to provide GUI features such as
D>rem scrollable output window and options to resize window and set
font, etc.
D>rem
D>prompt $P$G
D:\Work\eclipse\WORKSP~1\superzap\build\demos>zap demo10
D:\Work\eclipse\WORKSP~1\superzap\build\demos>java -classpath
"c:\program files\Automated Software Tools\SuperZap";"c:\program
files\Automated Software Tools\SuperZap\superzap.jar" superzap demo10
Timeout monitor for CMD and ZAP turned off
enter character to search for or q to exit
search key = 61
00000061 *61626364 65666768 696A6B6C 6D6E6F70* *abcdefghijklmnop*
enter character to search for or q to exit
search key = 78
00000078 *78797A7B 7C7D7E7F 80818283 84858687* *xyz{|}~.....*
enter character to search for or q to exit
exiting now
SuperZap total errors = 0
D:\Work\eclipse\WORKSP~1\superzap\build\demos>
SuperZap total errors = 0
```

# SuperZap User Guide V1.0.09 by Automated Software Tools

## 8.4 Release Notes:

- V1.0.05 - 01/12/05 initial release
- V1.0.06 - 01/13/05 update licensing
- V1.0.07 - 01/18/05 update to add pop-up if unsupported Java ver.
- V1.0.08 - 02/03/05 update shareware license
- V1.0.09 - 03/03/05 update to use open source GPL license