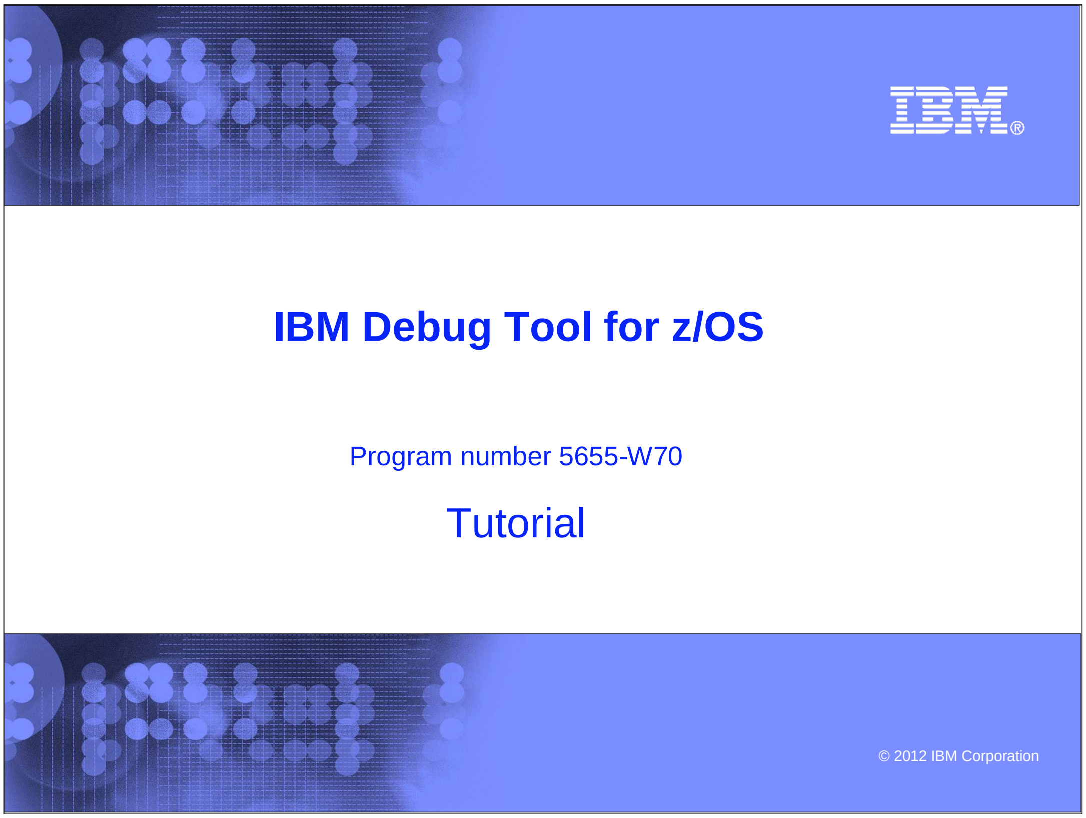
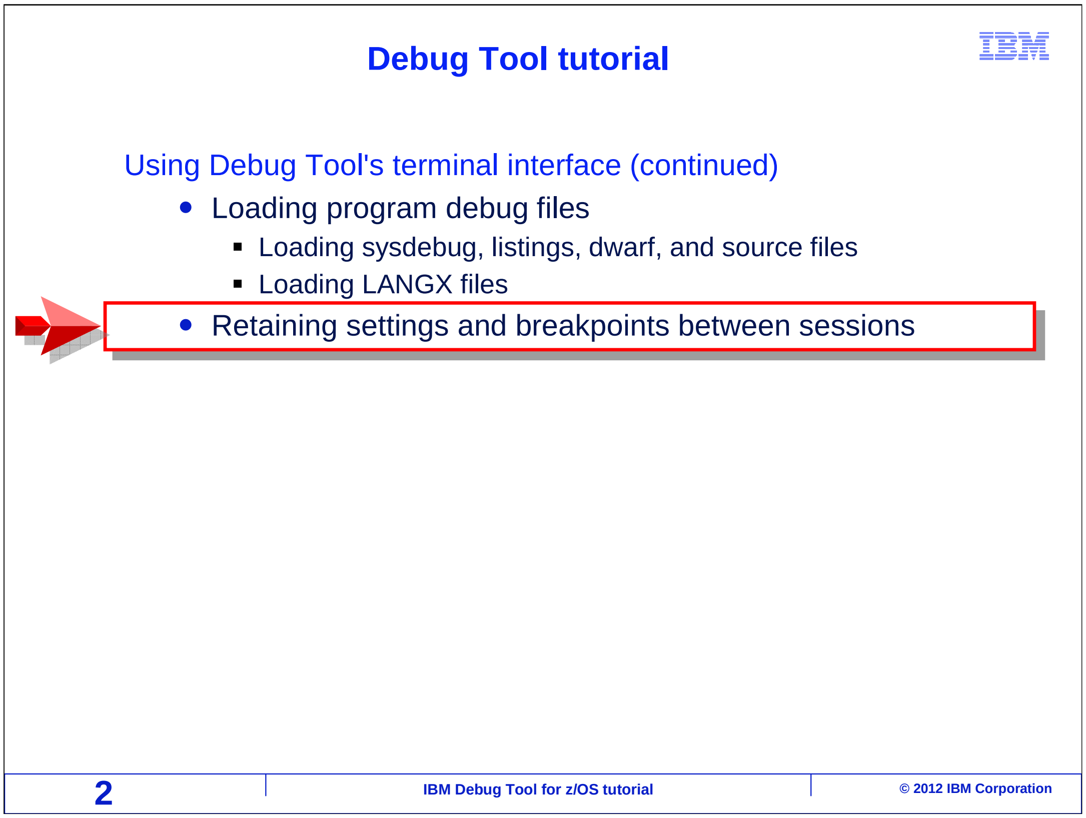


IBM Debug Tool for z/OS

Program number 5655-W70

Tutorial

This is the tutorial for IBM Debug Tool for z/OS®, one of the IBM zSeries® problem determination tools.

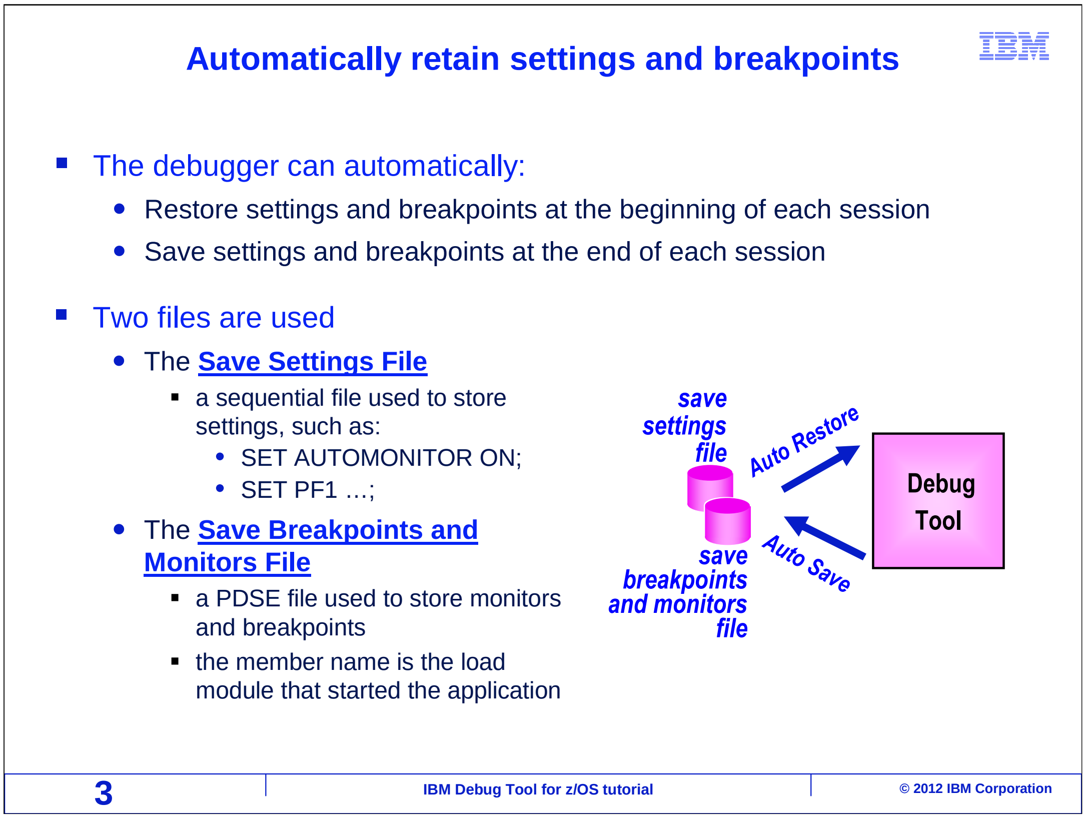


Debug Tool tutorial

Using Debug Tool's terminal interface (continued)

- Loading program debug files
  - Loading sysdebug, listings, dwarf, and source files
  - Loading LANGX files
- Retaining settings and breakpoints between sessions

In this section, you will learn how you can make Debug Tool automatically retain your settings, monitors, and breakpoints when you are using the terminal interface.



Automatically retain settings and breakpoints

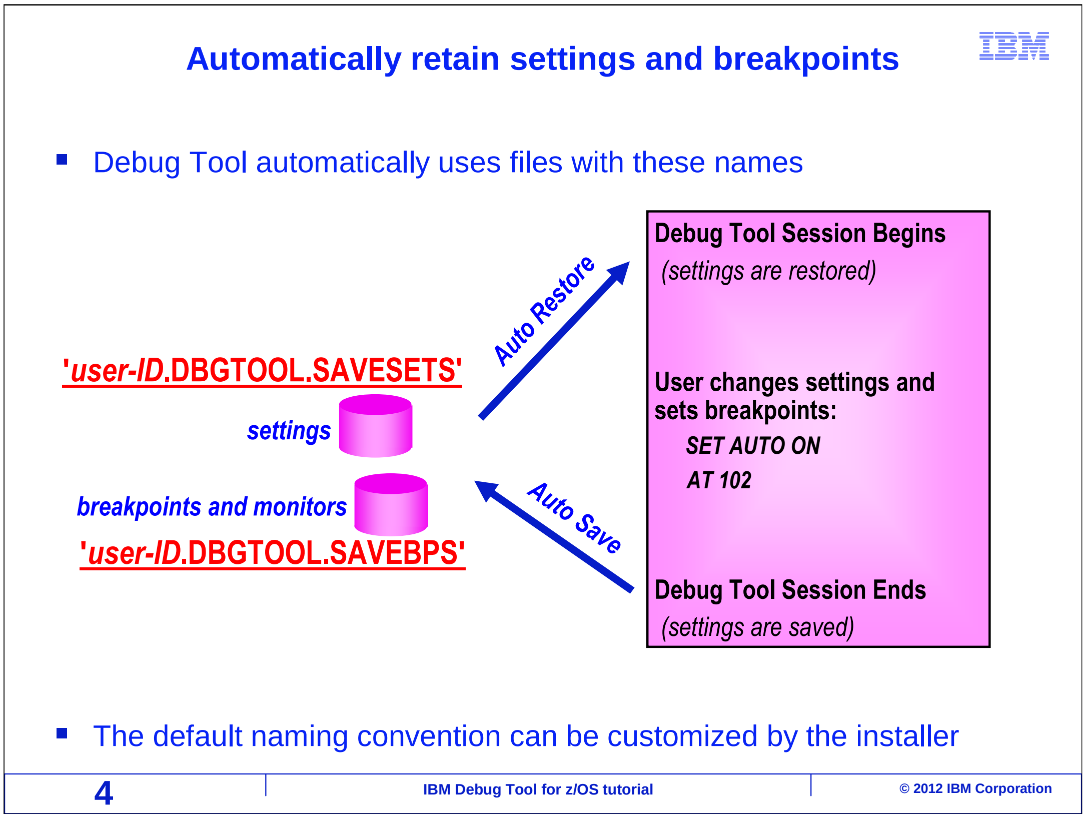
- The debugger can automatically:
  - Restore settings and breakpoints at the beginning of each session
  - Save settings and breakpoints at the end of each session
- Two files are used
  - The **Save Settings File**
    - a sequential file used to store settings, such as:
      - SET AUTOMONITOR ON;
      - SET PF1 …;
  - The **Save Breakpoints and Monitors File**
    - a PDSE file used to store monitors and breakpoints
    - the member name is the load module that started the application

3 IBM Debug Tool for z/OS tutorial © 2012 IBM Corporation

By default, when you are using the terminal interface, the debugger does not retain your settings and breakpoints between sessions. For example, say that you are debugging a program, and you turn on the auto monitor by entering the command “SET AUTO ON”. The auto monitor stays on for the rest of the session. But then, you Quit from the debugger, and after a while start a new session. The auto monitor will be off again, because your settings were reset to the defaults.

However, you can make the debugger automatically restore settings and breakpoints at the beginning of each session, and save settings and breakpoints at the end of each session. That way, your settings are permanent from session to session.

Two files are used. First, a “save settings file” is a sequential file that keeps settings, such as the auto monitor or function keys. Second, a “Save breakpoints and Monitors” library is used to keep your breakpoints and monitors. Unique members are kept for each main program, so each application can have it’s own set of breakpoints and monitors.



Automatically retain settings and breakpoints

- Debug Tool automatically uses files with these names

'*user-ID*.DBGTOOL.SAVESETS'

settings

breakpoints and monitors

'*user-ID*.DBGTOOL.SAVEBPS'

Debug Tool Session Begins
(settings are restored)

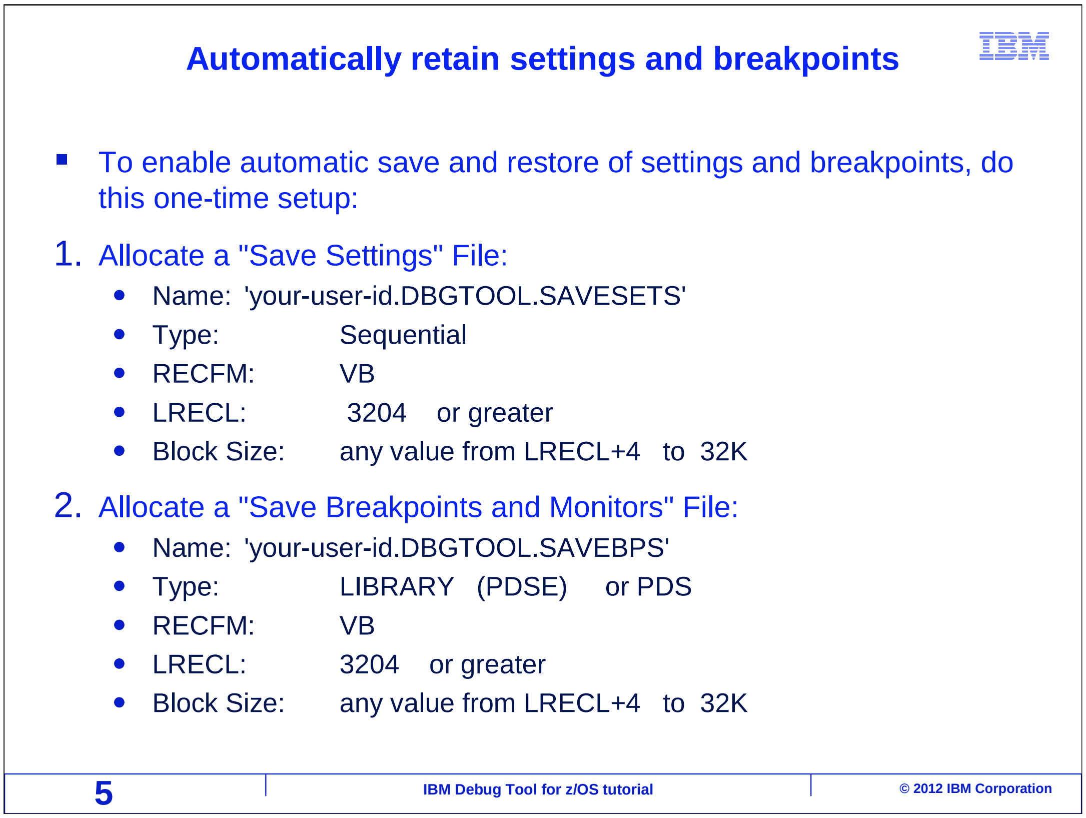
User changes settings and sets breakpoints:
SET AUTO ON
AT 102

Debug Tool Session Ends
(settings are saved)

- The default naming convention can be customized by the installer

By default, the debugger automatically uses files with the naming convention: user-ID.DBGTOOL.SAVESETS for the settings file, and user-ID.DBGTOOL.SAVEBPS for the breakpoints and monitors library. When the debugger starts, it can automatically look for files with these names, and if it finds them, it can restore the settings. When it ends, it can automatically write the current settings back to the files.

As an installation option, the person who installs Debug Tool can change this naming convention. Use files with these names unless you have been instructed otherwise by the installer.



Automatically retain settings and breakpoints

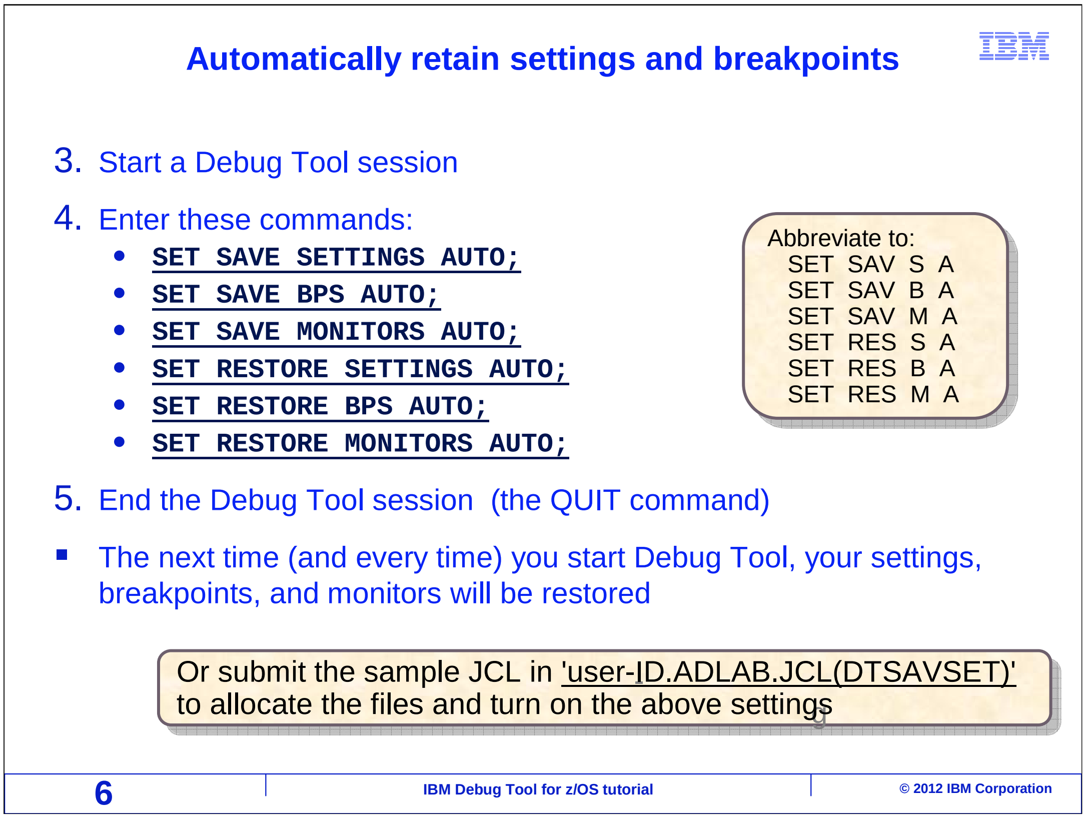
- To enable automatic save and restore of settings and breakpoints, do this one-time setup:

1. Allocate a "Save Settings" File:
   - Name: 'your-user-id.DBGTOOL.SAVESETS'
   - Type: Sequential
   - RECFM: VB
   - LRECL: 3204 or greater
   - Block Size: any value from LRECL+4 to 32K
2. Allocate a "Save Breakpoints and Monitors" File:
   - Name: 'your-user-id.DBGTOOL.SAVEBPS'
   - Type: LIBRARY (PDSE) or PDS
   - RECFM: VB
   - LRECL: 3204 or greater
   - Block Size: any value from LRECL+4 to 32K

5 IBM Debug Tool for z/OS tutorial © 2012 IBM Corporation

Here is how you configure the debugger to automatically save and restore your settings, breakpoints, and monitors. First, create your Save Settings file. Allocate it as a sequential file with a record format of VB (variable length blocked records), and a record length of 3204 or greater. The block size can be anything from the record length + 4 through 32K.

Then create your Save Breakpoints and Monitors file. Allocate it as a PDSE file with a record format of VB, and a record length of 3204 or greater. The block size can be anything from the record length + 4 through 32K. It will work if you make it a PDS, but it is better to make it a PDSE library.



Automatically retain settings and breakpoints

3. Start a Debug Tool session

4. Enter these commands:
   - **SET SAVE SETTINGS AUTO;**
   - **SET SAVE BPS AUTO;**
   - **SET SAVE MONITORS AUTO;**
   - **SET RESTORE SETTINGS AUTO;**
   - **SET RESTORE BPS AUTO;**
   - **SET RESTORE MONITORS AUTO;**

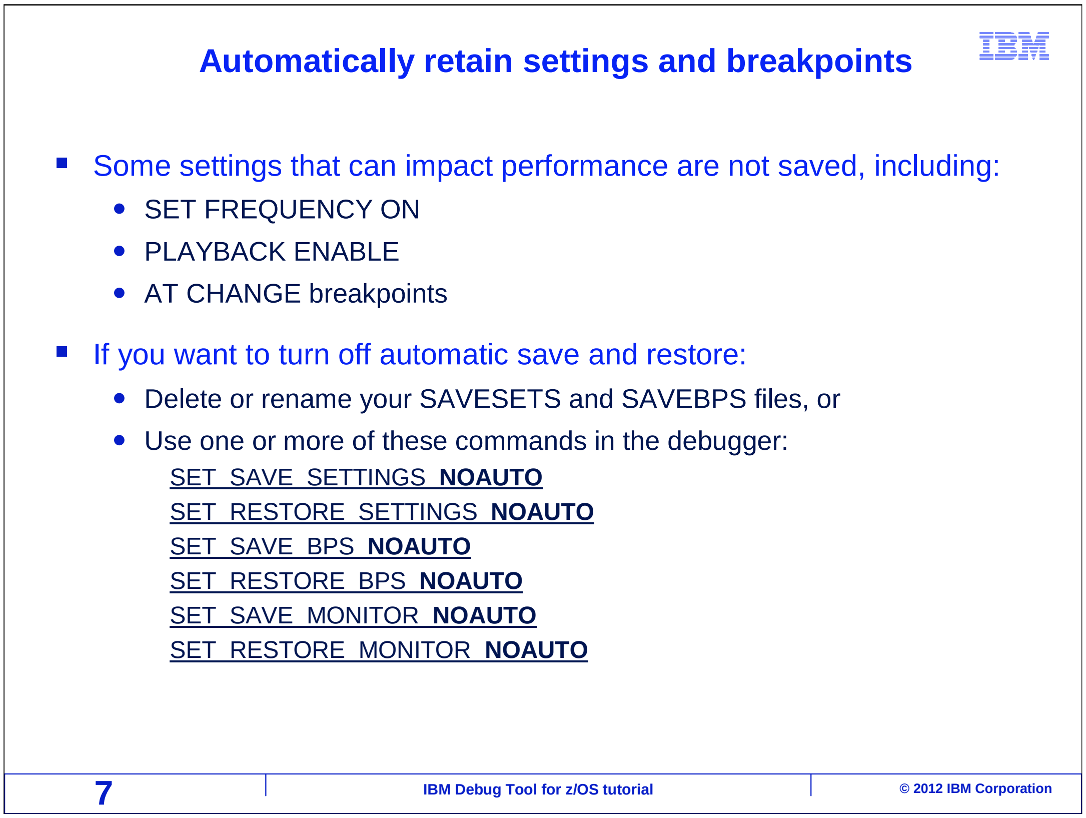
Abbreviate to:
SET SAV S A
SET SAV B A
SET SAV M A
SET RES S A
SET RES B A
SET RES M A

5. End the Debug Tool session (the QUIT command)

- The next time (and every time) you start Debug Tool, your settings, breakpoints, and monitors will be restored

Or submit the sample JCL in 'user-ID.ADLAB.JCL(DTSAVSET)' to allocate the files and turn on the above settings

Next, start a debugging session with any application. After it starts, enter the commands shown here. “SET SAVE” commands specify that the debugger should automatically save settings at the end of each session. “SET RESTORE” commands tell the debugger to automatically restore settings at the beginning of each session. After entering these commands, Quit from the debugger, and you are done. Then next time, and every time you use the debugger, your settings, breakpoints, and monitors will be restored and saved.



Automatically retain settings and breakpoints

- Some settings that can impact performance are not saved, including:
  - SET FREQUENCY ON
  - PLAYBACK ENABLE
  - AT CHANGE breakpoints
- If you want to turn off automatic save and restore:
  - Delete or rename your SAVESETS and SAVEBPS files, or
  - Use one or more of these commands in the debugger:

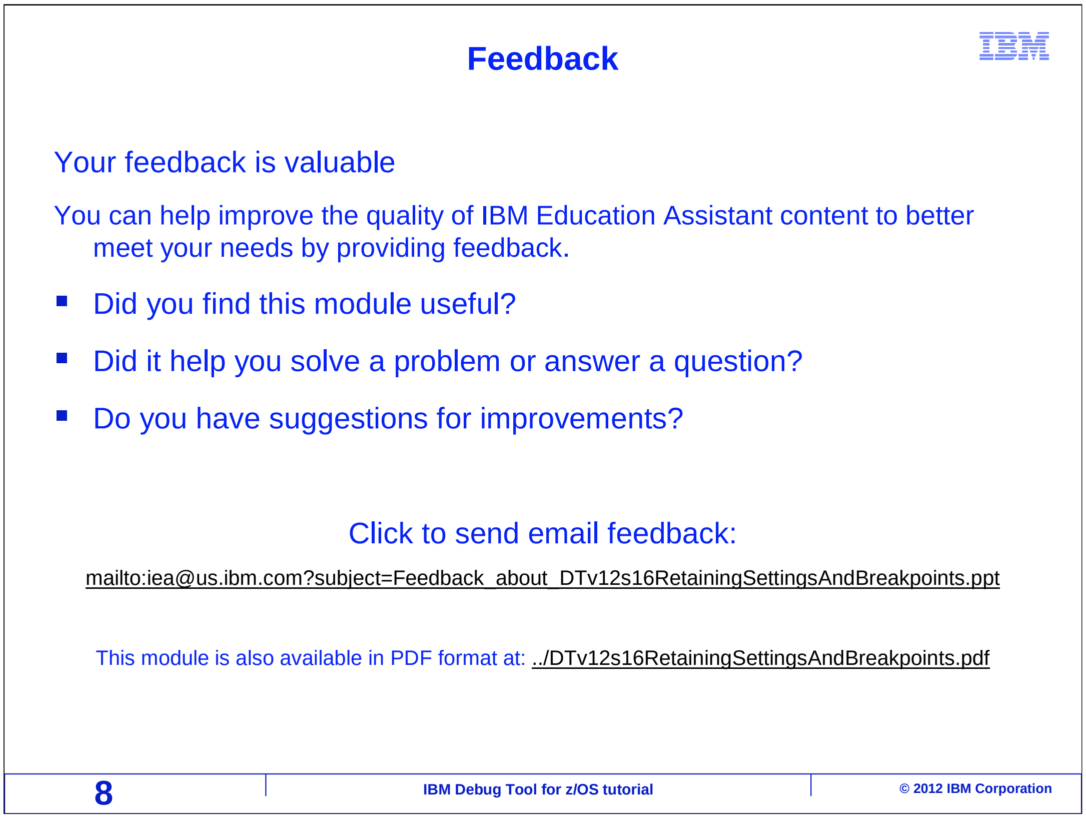
    SET SAVE SETTINGS **NOAUTO**
    SET RESTORE SETTINGS **NOAUTO**
    SET SAVE BPS **NOAUTO**
    SET RESTORE BPS **NOAUTO**
    SET SAVE MONITOR **NOAUTO**
    SET RESTORE MONITOR **NOAUTO**

7 IBM Debug Tool for z/OS tutorial © 2012 IBM Corporation

If at a later point you decide that you do not want to automatically save and restore, there are a couple of ways to turn it off. If you delete or rename your Save Settings File, the debugger will not be able to find it to restore your settings. Or the commands “SET SAVE SETTINGS NOAUTO” and “SET RESTORE SETTINGS NOAUTO” will turn it off.

You can have the debugger retain settings, but not breakpoints or monitors. Commands such as SET RESTORE BPS NOAUTO and SET RESTORE MONITOR NOAUTO selectively turns off automatic retention of breakpoints or monitors.

That is the end of this section, automatically saving and restoring settings, breakpoints, and monitors when you are using the terminal interface.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

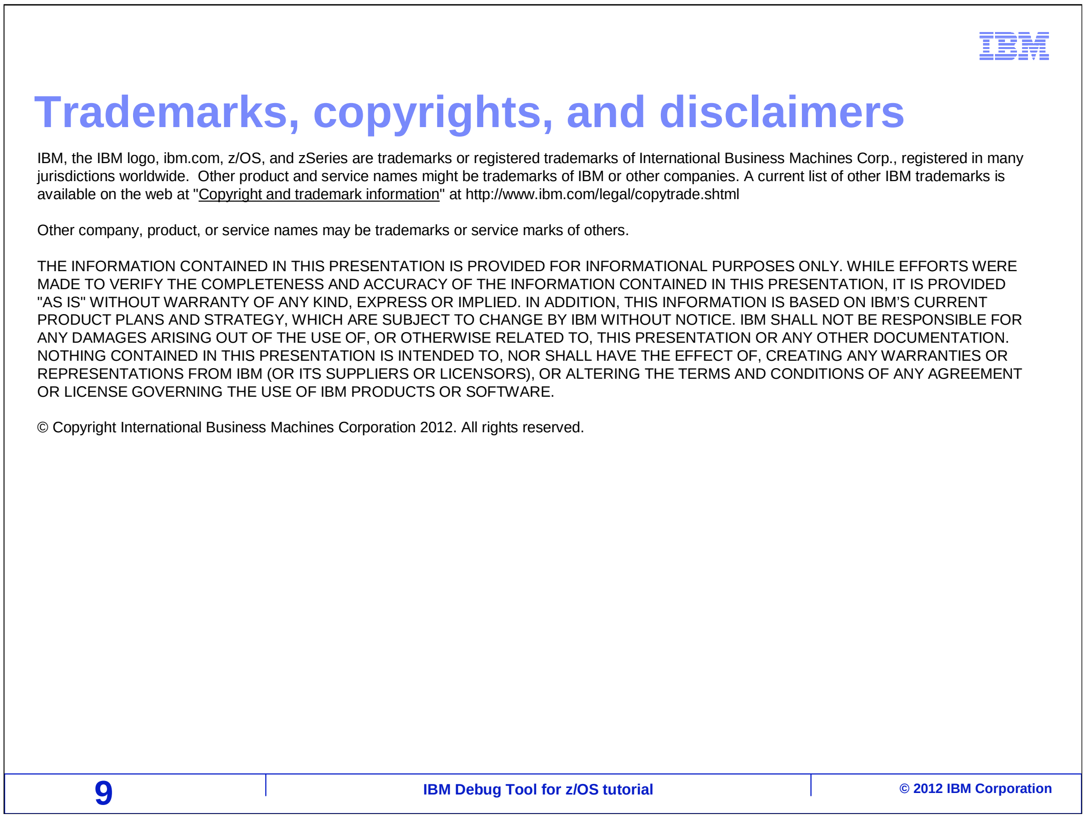
Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_DTv12s16RetainingSettingsAndBreakpoints.ppt

This module is also available in PDF format at: ../DTv12s16RetainingSettingsAndBreakpoints.pdf

8 IBM Debug Tool for z/OS tutorial

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, z/OS, and zSeries are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.