Lab 7 – Problem Determination

At the end of this exercise, you should be able to:

- Use the administrative console to configure and view log data
- Enable a server to use HPEL
- Enable tracing on application server components
- Use the HPEL Log Viewer to examine log and trace data
- Enable verbose garbage collection for an application server
- Enable memory leak detection for an application server
- Describe how IBM Support Assistant tools can be used to analyze JVM memory dumps

Section 1: Working with log files of the application server

In this section, you examine the configuration options for logging in Basic mode.

Section 2: Set up and configure HPEL

High Performance Extensible Logging (HPEL) is a new mode of logging and tracing. To take advantage of this new log and trace framework, HPEL mode must be enabled. After HPEL mode is enabled, the JVM logs (typically `SystemOut.log` and `SystemErr.log`), the trace log (typically `trace.log`), and the service log (typically `activity.log`) are no longer written to. Instead, log and trace content is written to a log data or trace data repository in a proprietary binary format and, if configured, to a text log file. By disabling the text log file, you gain the largest possible performance benefit of HPEL. A log viewing tool, Log Viewer, is provided to allow for viewing, filtering, monitoring, and formatting the log and trace data in the repositories.

In this section, you enable HPEL mode for server1. Then, you explore and modify the log and trace configurations.

Section 3: Use the Log Viewer in the administrative console to examine log data and trace data

In this section, you use the Log Viewer in the administrative console to examine the log messages for an application server. You use various filtering functions to customize what log records are shown.

Section 4: Enable tracing for an application server and view trace data from the Log Viewer

In this section, you configure tracing on the session management components of server1. Use the PlantsByWebSphere application to generate trace data, and view the trace data in the Log Viewer.

Section 5: Enable cross-component trace (XCT)

In this section, you learn how to enable cross-component trace (XCT) for an application server. You also examine the request IDs and other data that XCT provides in the server logs.

Include request IDs and trace records and create correlation log records

This setting enables XCT to include request IDs in log and trace files when you want to see which log and trace entries, in all threads and application server processes, are related to the same request. Request IDs are recorded only when using HPEL log and trace mode and can be seen or used for filtering when using the `logViewer` command.

In addition, XCT creates correlation log records when you want to log how requests branch between threads and processes, and see extra information about each request.

Warning: Enabling XCT to create correlation log records might have a significant performance cost on your system, so is best suited to test and development environments.

IBM Cross Component Trace Log Viewer

Available in the IBM Support Assistant, IBM WebSphere Cross Component Trace Log Viewer provides enhanced log file views for logs that are augmented with Cross Component Trace correlation log records. Logs can be displayed in flat or hierarchical layouts, and multiple logs can be loaded and viewed simultaneously with log entries related to each request conveniently grouped.

Section 6: Collecting JVM data

There are several common JVM-related problems such as hung threads, memory leaks, and out-of-memory conditions. This section shows you how to collect diagnostic data to help troubleshoot these problems. First, you install an example application that is written to illustrate several JVM-related problems.

Section 7: Clean up server1

The last section concludes the active exercise. The next section is read-only. Follow these steps to clean up server1 and uninstall the BadApp application.

Section 8: READ ONLY: Using IBM Support Assistant tools to analyze JVM data