

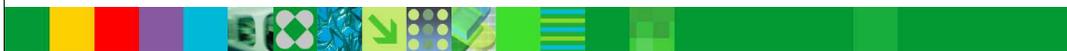


IBM Software Group

Content Manager OnDemand

ODWEK – Enabling tracing, logging and requirements

IBM Information Management software



@business on demand.

© 2009 IBM Corporation
Updated June 26, 2009

This presentation covers enabling diagnostics and troubleshooting ODWEK.

Purpose

- Provide an overview of enabling ODWEK logging and tracing analysis and ODWEK requirements

The purpose of this presentation is to provide an overview of enabling ODWEK logging and tracing analysis and ODWEK requirements.

Enabling ODWEK tracing and logging

- In ODWEK V7.1.2.5, a new trace mechanism was introduced
 - ▶ Previous versions used logging (arswww.log)
- ODWEK trace file is archived and renamed when ODWEK application is restarted
 - ▶ ODWEK Java™ APIs will archive and rename the arswwww.trace file if all ODServer objects are currently terminated
 - ▶ Archived and renamed to arswwww.trace<timestamp>

A new trace mechanism was introduced in ODWEK version 7.1.2.5. Previous versions used ODWEK logging.

When the ODWEK application is restarted, the ODWEK trace file is archived and renamed. The ODWEK Java™ APIs will archive and rename the trace file if all ODServer objects are currently terminated. When ODWEK archives the trace file, the timestamp is appended to the arswwww.trace file name.

Enabling ODWEK tracing and logging (continued)

- Enabling ODWEK tracing will have a noticeable performance impact
- ODWEK trace file not maintained and can grow very large (hundreds of MBs, GBs)

As with any trace, enabling ODWEK tracing will have a noticeable impact on performance. Since the ODWEK trace file is not maintained, the file size can grow very large, possibly hundreds of MBs, or GBs.

Enabling ODWEK tracing and logging (continued)

- Enabling ODWEK tracing (ODWEK V7.1.2.5 or later)
 - ▶ Instructions to enable:
http://www-1.ibm.com/support/docview.wss?rs=129&context=SSEPCD&context=SSEPC6&q1=odwek+trace&uid=swg21240220&loc=en_US&cs=utf-8&lang=en
 - ▶ If using the ODConfig object in the Java APIs, set the TraceLevel parameter accordingly (values of 1-4)

To enable ODWEK tracing when using ODWEK V7.1.2.5 or later, follow the instructions on the “How do I enable DB2® Content Manager OnDemand Web Enablement Kit (ODWEK) trace?” page. The address is displayed on this slide. If using the ODConfig object in the Java APIs, set the TraceLevel parameter accordingly (values of 1-4).

Enabling ODWEK tracing and logging (continued)

- Enabling ODWEK logging (ODWEK V7.1.2.4 or earlier)
 - ▶ Add/uncomment this debug stanza
 - [debug]
 - log=1
 - logdir=/ars/tmp
 - ▶ The log file, arswwww.log is written into the directory referenced by 'logdir'
 - ▶ For ODWEK servlet implementations, you will need to restart the servlet for the arswwww.ini change to be reflected

To enable ODWEK logging when using ODWEK V7.1.2.4 or earlier, add or uncomment the debug stanza so it reflects the code as shown on the slide. The log file, arswwww.log, is written into the directory referenced by 'logdir'. For ODWEK servlet implementations, you will need to restart the servlet for the arswwww.ini to see the change.

ODWEK requirements: arswww.ini

- Each instance of ODWEK in the same environment, must reference its own:
 - arswww.ini
 - CacheDir
 - TempDir
 - TraceDir
- If using ODConfig object instead of arswww.ini, the TempDir and TraceDir should be unique per ODWEK application

For ODWEK requirements for arswww.ini, each instance of ODWEK (CGI, servlet, or Java API) in the same environment must reference its own unique arswww.ini, CacheDir, TempDir, and TraceDir parameter values.

For example, if a customer has two ODWEK Java API applications deployed in WebSphere® on the same machine, each ODWEK application must have its own unique arswww.ini, CacheDir, TempDir, and TraceDir.

To avoid unexpected results, a customer using the ODConfig object should have a unique TempDir and TraceDir per ODWEK application.

ODWEK requirements: installation or upgrade

- ODWEK (CGI, servlet, or Java API) must be installed using the InstallShield or native install packages
 - ▶ Copying ODWEK files from one machine to another is not supported
 - ▶ Install using root or Windows® Administrator account
- If ODWEK CGI or servlet was upgraded
 - ▶ CacheDir and TempDir contents need to be deleted following an upgrade

In regards to ODWEK installation and upgrade requirements, ODWEK (CGI, servlet, or Java API) must be installed using the InstallShield or native install packages. Copying ODWEK files from one machine to another is not supported. You must install using root or a Windows Administrator account.

If ODWEK CGI or servlet was upgraded, the CacheDir and TempDir contents need to be deleted in order to avoid unexpected results, or at worse, an ODWEKabend.

ODWEK requirements: Native heap

- ODWEK Servlet and Java API native heap requirement
 - ▶ $2 * (\text{average uncompressed document size}) * (\text{average number of concurrent document retrievals})$
- ODWEK Servlet and Java API Java heap requirement
 - ▶ $(\text{average uncompressed document size}) * (\text{average number of concurrent document retrievals})$

In general, the ODWEK native heap requirement is two times the average uncompressed document size multiplied by the average # of concurrent document retrievals.

Also, in general, the ODWEK Java heap requirement is the average uncompressed document size multiplied by the average number of concurrent document retrievals.

ODWEK requirements: Native heap (continued)

- ODWEK requires more native heap than Java heap
- ODWEK will abend if unable to allocate enough contiguous native heap for the requested operation
- Memory not allocated or reserved to Java heap is available as native heap
 - ▶ JVM reserves the maximum Java heap (Xmx) on initialization

ODWEK is not a pure Java application, most of ODWEK's processing is performed in the native C shared library (libarswwwsl or ars3wapi). For this reason, ODWEK requires more native heap than Java heap.

If ODWEK cannot allocate enough contiguous native heap for the requested operation, ODWEK will typically abend. To avoid an abend, ensure that the Java maximum heap (Xmx) is not set too high. Memory not allocated or reserved to Java heap is available as native heap. On initialization, the JVM allocates the entire Java heap in a single contiguous area of virtual storage. The amount that is allocated is determined by the setting of the -Xmx parameter. No virtual space from the heap is ever freed back to the native operating system. When the heap shrinks, it shrinks inside the original virtual space.

ODWEK requirements: Native heap (continued)

- A single process can allocate up to 4GB of memory (2^{32} address space) in a 32-bit environment
 - ▶ Varying amounts of this 4GB address space is reserved for the user process (Java) and kernel
 - AIX® – user process can allocate up to 2.5GB (configurable up to 3.25GB)
 - Linux® - OS reserves 1GB for kernel and 3GB for user process
 - Windows - OS reserves 2GB for kernel and 2GB for user process
 - Solaris – 3GB for user process (depends on how much physical memory)

In a 32-bit environment, the maximum amount of memory a single process can allocate is 4GBs, but depending on the OS, this 4GB address space is shared between the user process and the kernel. This leaves much less memory for the user process, Java, for Java heap and native heap allocations.

For example, the maximum amount of memory a user process can allocate when using AIX is 2.5GBs, which can be configured up to 3.25GBs. For Linux, the OS reserves 1GB for kernel and allows 3GB for the user process. The OS reserves 2GBs for kernel and allows 2GBs for the user process in a Windows environment. And in general for Solaris, a 3GB address space can be allocated by a user process; however this depends on how much physical memory exists.

Another example is if a 32-bit environment has 10GBs of virtual memory. A single process can only allocate a theoretical 4GB amount. To use more virtual memory, either a 64-bit environment should be considered or running a clustered application server environment. This will allow multiple Java processes to be launched, each independently hosting the same ODWEK application.

ODWEK requirements: Native heap (continued)

- Document retrievals are the most native memory intensive operations
 - ▶ ODWEK requires native heap for the entire document to be in memory for processing
 - ▶ ODWEK was not designed to be used to retrieve very large documents (several hundred MB)
 - Large documents can be archived into OnDemand with Large Object support instead.
 - ▶ There are ODWEK Java APIs to retrieve directly to file instead (no native heap is used)
 - ODHit.retrieve (viewer, file name)
 - ODHit.retrieveSegment (segment, file name)
 - ODHit.getDocument (file name, allSegs)

12

ODWEK Enabling tracing, logging and requirements

© 2009 IBM Corporation

Document retrievals are the most native memory intensive operations, as ODWEK requires native heap for the entire document to be in memory for processing. ODWEK was not designed to be used to retrieve very large documents, for example, several hundred megabytes. Instead, it is recommended to load large documents under Large Object support which is only supported with an ACIF, OS/390®, or OS/400® indexers. Large Object breaks the document into logical segments, which are retrieved on the fly as needed by ODWEK.

For the ODWEK Java APIs, there are APIs to retrieve documents directly to file as well. No native heap is used.

ODWEK requirements: Native heap (continued)

- AFP resources are also cached in native heap compressed
 - ▶ ODWEK caches unique AFP resources per application group
 - ▶ Not recommended to use large AFP resources (larger than a few MB)
 - ▶ For ODWEK Java API, only occurs when ODHit.retrieve API is used to retrieve documents

ODWEK servlet and Java APIs cache retrieved AFP resources, compressed, in native heap. Unique AFP resources are cached on a per application group level. It is not recommended to utilize large AFP resources for this reason – memory requirements will be higher. In general, large AFP resources are not optimal for storage, memory, and document print and or retrieval performance in OnDemand.

If using the ODWEK Java APIs, AFP resource caching will only occur if the ODHit.retrieve API is used to retrieve documents.

ODWEK requirements: Native heap (continued)

- AFP resources are freed from native heap in the following situations:
 - ▶ When the ODWEK servlet is stopped
 - ▶ When using the ODWEK Java API, all ODServer objects are terminated at a given time (ODServer.terminate)
- ODWEK Java APIs provide an API (ODHit.getResources) to implement your own AFP resource caching scheme instead

AFP resources are freed from native heap in the following situations:

-When the ODWEK servlet is stopped.

-When using the ODWEK Java API, all ODServer objects are terminated at a given time (ODServer.terminate).

If you want to implement your own AFP resource caching scheme instead, the ODWEK Java APIs provide an API to retrieve resources ODHit.getResources. The ODHit.getDocument API can then be used to retrieve just the document data. Concatenating the AFP resource and document data together will form the entire AFP document.

ODWEK requirements: Native heap (continued)

- When more memory is required, ODWEK Java APIs can be upgraded to ODWEK V8.4 on UNIX®
 - ▶ ODWEK can be run in 64 bit mode (requires 64 bit application server and OS)
 - ▶ ODWEK Java APIs are only provided as 32 bit on Windows
 - ▶ ODWEK servlet and CGI can only be run in 32 bit mode on any platform

15

ODWEK Enabling tracing, logging and requirements

© 2009 IBM Corporation

If further memory is required, customers using the ODWEK Java APIs can upgrade to ODWEK V8.4 on UNIX as ODWEK can be run in 64 bit mode. ODWEK V8.4 requires a 64 bit application server and OS.

ODWEK Java APIs are only provided as 32 bit on Windows and ODWEK servlet and CGI can only be run in 32 bit mode on any platform.

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 e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_ODWEK_LoggingTracing2.ppt

This module is also available in PDF format at: [../ODWEK_LoggingTracing2.pdf](http://ODWEK_LoggingTracing2.pdf)



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

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, [ibm.com](http://www.ibm.com), and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX DB2 OS/390 OS/400 WebSphere

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. 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>

Windows, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java, JVM, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

