Compaq

White Paper

April 2002

Prepared by:

Global SAP Solutions
Compaq Computer Corporation

Edition 1

Contents

APO 3.0 and liveCache Parameterization in Tru64 Environment

Abstract: This document provides information about SAP APO support packages and liveCache database configuration on Tru64. The information is presented in tables that show the relationship among the different elements in an APO/liveCache environment.

SAP-ICCEngineering@compaq.com

Notice

The information in this publication is subject to change without notice and is provided "AS IS" WITHOUT WARRANTY OF ANY KIND. THE ENTIRE RISK ARISING OUT OF THE USE OF THIS INFORMATION REMAINS WITH RECIPIENT. IN NO EVENT SHALL COMPAQ BE LIABLE FOR ANY DIRECT, CONSEQUENTIAL, INCIDENTAL, SPECIAL, PUNITIVE, OR OTHER DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, OR LOSS OF BUSINESS INFORMATION), EVEN IF COMPAQ HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The limited warranties for Compaq products are exclusively set forth in the documentation accompanying such products. Nothing herein should be construed as constituting a further or additional warranty.

This publication does not constitute an endorsement of the product or products that were tested. The configuration or configurations tested or described may or may not be the only available solution. This test is not a determination of product quality or correctness, nor does it ensure compliance with any federal, state or local requirements.

Compaq, NonStop, Deskpro, Compaq Insight Manager, Systempro, Systempro/LT, ProLiant, ROMPaq, QVision, SmartStart, NetFlex, QuickFind, PaqFax, and Prosignia are registered with the United States Patent and Trademark Office.

ActiveAnswers, Netelligent, Systempro/XL, SoftPaq, Fastart, QuickBlank, QuickLock are trademarks and/or service marks of Compaq Computer Corporation.

Microsoft, Windows and Windows NT are trademarks and/or registered trademarks of Microsoft Corporation.

The following are trademarks or registered trademarks of SAP AG; ABAP/4, InterSAP, RIVA, R/2, R/3, R/3 Retail, SAP (Word), SAPaccess, SAPfile, SAPfind, SAPmail, SAPoffice, SAPscript, SAPtime, SAPtronic, SAP-EDI, SAP EarlyWatch, SAP ArchiveLink, SAP Business Workflow, and ALE/WEB. The SAP logo and all other SAP products, services, logos, or brand names included herein are also trademarks or registered trademarks of SAP AG.

Intel, Pentium and Xeon are trademarks and/or registered trademarks of Intel Corporation.

Oracle is a registered trademark of Oracle Corporation.

Other product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

APO 3.0 and liveCache Parameterization in Tru64 Environment
White Paper prepared by Global SAP Solutions

Edition 1 (April 2002)

Introduction

The aim of this document is to provide information about SAP APO Support Packages and liveCache database configuration on Tru64. Most of the information is presented in tables that show the relationships among the different elements in an APO/liveCache environment.

A feature of this document is that it lists the number of the SAP OSS Notes that are related to every important concept mentioned. If, for example, you are looking for a liveCache parameter, you will find the number of the notes where that parameter is mentioned in the liveCache's parameters table.

This document also provides important information about the sizing of the memory in Tru64 when Transactional Simulation is used (for instance with /SAPAPO/OM_PERFORMANCE report) for nonproductive systems.

APO Support Packages

The different packages, patches and libraries mentioned in Table 1 can be found at the URLs below:

- Support Packages
  http://service.sap.com/ocs-download

- Binary Patches and Support Packages
  http://service.sap.com/swcenter-main

- APO COM Libraries

  ftp://sapserv<X>/specific/apo/abap/sapinternal/qm/support/apo_30/TRU64/ (support package < 19)

  ftp://sapserv<X>/specific/apo/apo30/<SPn>/COM/COMPAQ_TRUE64/SAPCOM30_<build number>_TRU_<change list number>.SAR (support package ≥ 19)

- liveCache Upgrade
  ftp://sapservX/general/sapdb/LC_VERSIONS/<version>/DEC_64/sapdb-server-tru64-64bit-alpha-<version>.sar

Support Packages for APO Release 3.0A

SAP Note 303743

This table shows the relationship among the different versions of Support Packages and applications related to the installation of an APO support package.

The columns have the following meanings:

1. Item (SP/App): Support Packages, applications or libraries.
2. N°: number of the Support Packages or version of the application/library.

3. Notes: number of the OSS Notes that reference the SP or application/library in the left column.

For instance, for APO SP15 you need Business and Information Warehouse SP17 (see first column "N^o"). Only the latest two APO SP numbers are in the table.

The order in which the SP must be installed is shown with (0), (1), (2), etc. before the name of the SP.

Item (SP/App)	N^o	Notes	N^o	Notes	N^o	Notes
(0) SPAM	Latest	97622	Latest	97622	Latest	97622
(1) Basis – KB (4.6C)	≥23	104664	≥23	104664	≥28	104664
(2) ABA – KA (4.6C)	≥23	104664	≥23	104664	≥28	104664
(3) BW – KW (2.0B)	≥17	328237	≥20	328237	≥20	328237
(4) APO – KY (3.0A)	17	448518	18	459985	19	491882
SAP GUI	-	422446	-	422446	-	422446
LiveCache	7.2.5.14		7.2.5.14		7.2.5.15	379051
COM libraries build	27	459658	27	459658	28	492320
R/3 Kernel	4.6D		4.6D		4.6D	
R/3 patch level[1]	≥579	19466	≥579	19466	≥579	19466
Dbadaslib[2]	Latest	325402	Latest	325402	Latest	325402
RFC connection	-	305634	-	305634	-	305634

Table 1, Support Package Overview

Installation steps:

1. Log on as user: <sid>adm.

2. In the transport system of your APO system change to subdirectory trans.

3. Unpack the archive which contains the patches with the statement:
    CAR -xvf <path>/<archive>.CAR

4. Log on to your APO system in client 000 as a user with SAP_ALL authorizations. Do NOT use the users SAP* or DDIC.

5. Display the patches in the R/3 system: transaction SPAM → Support Package → Load packages → From application server.

---

[1] dw1_<patch>.CAR, dw2_<patch>.CAR

[2] dw1_<patch>.CAR, dw2_<patch>.CAR

6. Apply the patch. Call Transaction SPAM first. For other patch types, enter the long name in the input field or determine the patch queue with function key F4. You can then continue with the application procedure as usual with the 'Apply' button.

After importing an APO Support Package, it is advisable to run Transaction **SGEN**. If you import several support packages one after another, you only have to do this once after the last support package has been imported.

1. Go to Transaction **SGEN**.

2. Choose the option 'Regeneration of existing loads'.

3. Choose the option 'only generate objects with invalid loads'.

This procedure ensures that the objects changed by the support package become new.

For more information, please refer to the documentation in the initial screen of Transaction **SGEN**.

APO COM Libraries

Note 326494, 391746

Table 2 provides information on the build of the COM libraries. In Table 1, the COM build is mentioned in connection with the APO SP and/or liveCache version. Here you can verify which "COM library change list" correspond to each APO Support Package.

You can see the "change list" of each APO COM build with transaction /SAPAPO/OM04 in an APO system. Please refer to http://service.sap.com/scm -> Folder "SCM Technology" -> Folder "Availability of SP and COM Builds" -> Document "Overview Matrix APO 3.0A SP/COM/liveCache Versions" or "Overview Matrix APO 3.1 SP/COM/liveCache Versions".

COM Build	COM change list	APO Support Package	liveCache
21	278853	13	7.2.5.4
22	280321	14	7.2.5.7
23	282334	15	7.2.5.7
24	284305	16	7.2.5.7
27	286575	17	7.2.5.14
27	288430	18	7.2.5.14
28	290043	19	7.2.5.15

Table 2, COM Library Builds

LiveCache Parameter Values

Note 388610

LiveCache parameters are divided into three sections: General, Extended and Support.

Some of these are included here together with an explanation about each one. The "Description" column contains the following information for each parameter:

- Version: the latest liveCache version where this parameter appears at the time this document was written.

- Section: parameter section.

- Value: default value or recommended value in the OSS Notes. The "Tru64" value that applies for this parameter is shown here too. This value comes from the APO Mass Test tests and is not intended to be unique or tuned for a real APO production system. It is nevertheless provided as a hint.

- Notes: number of OSS Notes that mention this parameter.

- Description: what the parameter means.

The most important parameters that apply to the APO Mast Test and have to be considered in an APO system are: DATA_CACHE, MAXCPU, MAXLOCKS, OMS_HEAP_LIMIT.

The way to show and/or change liveCache parameters from the dbmcli command line interface is:

> #> dbmcli –d <name> -n <server> -u control,control
>
> #> param_direcget
>
> #> param_directput <parameter> <value>
>
> #> param_checkall

Parameter	Description	
_USE_PROC_SCOPE	Version	7.2.5.7
	Section	Support
	Value	Yes Tru64 Yes
	Notes	427643, 424886, 433115
	Description	Defines which thread category is used for liveCache threads. This parameter must be set to YES when _USE_TRHD_FOR_TSK = YES.
_USE_THRD_FOR_TSK	Version	7.2.5.7
	Section	Support
	Value	Yes Tru64 Yes
	Notes	427643, 424886, 433115
	Description	Defines whether each liveCache task runs in an own operating system thread. If this parameter is not set to YES the liveCache crashes with signals 6, 10, 11. There is a message in knldiag file: DBSTATE kernel exit code: n, where n is 6,10,11,134,138,139.
_MAXTASK_STACK	Version	7.2.5.7
	Section	Support
	Value	1500 KB <= _MAXTASK_STACK <= 4096 KB Tru64 1500 KB
	Notes	433115
	Description	The value _MAXTASK_STACK * MAXUSERTASKS is the memory occupied by the liveCache tasks in KB.
CAT_CACHE_SUPPLY	Version	7.2.5.7
	Section	Extended
	Value	(MAXUSERTASKS + 1) * _CAT_CACHE_MINSIZE / 8192 Tru64 1632 Pages (8KB) → (50 +1) * 262144 / 8192
	Notes	433115
	Description	Maximum size in 8 KB pages of the cache for liveCache catalog information. The best performance can be achieved at a catalog cache hit rate of 100%. Use the database monitoring to get information about the catalog cache hit rate.

Parameter	Description	
DATA_CACHE	Version	7.2.5.7
	Section	General
	Value	50% RAM
	Notes	433115
	Description	Depends on the current or planned data volume (sizing). Unit = 8 KB pages. If no statements regarding the data volume are available yet, we recommend an initial value that is 50% of the physical memory (64Bit server) for a dedicated liveCache server.
DIAG_HISTORY_NUM	Version	7.2.5.7
	Section	Extended
	Value	2 Tru64 2
	Notes	433115
	Description	Number of the crash histories (logs)
DIAG_HISTORY_PATH	Version	7.2.5.7
	Section	Extended
	Value	<rundirectory>/DIAGHISTORY Tru64 default
	Notes	433115
	Description	Directory in which crash histories (dumps, diagnosis files, and so on) are being saved. Size required KB = DIAG_HISTORY_NUM * (2 * DATA_CACHE * 8 + OMS_HEAP_LIMIT) KB
KERNELDIAGSIZE	Version	7.2.5.7
	Section	Extended
	Value	100 Pages (8KB) Tru64 100
	Notes	433115
	Description	Size of the file for liveCache messages. You might have to enlarge it within the framework of diagnosis activities.
MAXARCHIVELOGS	Version	7.2.5.7
	Section	General
	Value	2 Tru64 2
	Notes	433115
	Description	Number of physical archive logs devices.

Parameter	Description	
MAXBACKUPDEVS	Version	7.2.5.7
	Section	General
	Value	2 Tru64 2
	Notes	433115
	Description	Maximum number of backup media that can be used in parallel with a liveCache backup.
MAXCPU	Version	7.2.5.7
	Section	General
	Value	> 1 Tru64 4 liveCache + other applications → MAXCPU = (number of processors) * 2 / 3
	Notes	410002, 425051, 433115
	Description	Maximum number of processors used by the liveCache for processing application requests (SQL, COM routines). How many processors active users in the liveCache are using simultaneously. By controlling the number of processors used in parallel, you can improve the scaling. If more applications are used on the liveCache server (APO, DB) then restrict MAXCPU to the maximum number of processors to be used by the liveCache on the basis of a detailed sizing. In case of blocks through long-running COM routines change the liveCache configuration and thus MAXCPU in accordance with Note 425051.
MAXDATADEVSPACES	Version	7.2.5.7
	Section	General
	Value	Maximum size of all DevSpaces + reserve.
	Notes	433115
	Description	Maximum number of data devspaces.
MAXDATAPAGES	Version	7.2.5.7
	Section	General
	Value	(Current size of all DevSpaces) * 1,25 Pages (8KB)
	Notes	433115
	Description	(Number of data DevSpaces) + 1.Depending on the current and planned data volume (sizing).

Parameter	Description	
	Version	7.2.5.7
	Section	General
	Value	50000 ≤ MAXLOCKS ≤ 350000 Tru64 50000
	Notes	433115, 337773, 65946
MAXLOCKS	Description	Maximum number of SQL locks. Maximum number of lock list entries in which the current and requested row and table locks for all users are held. Sets the maximum size of the lock list in which the record or table locks of all users as well as their lock requests are managed. Use the database monitoring (st04 → Detail analysis menu → Exclusive lock waits) to obtain information about the number of Max. Entries, Escalations, and Average Used Entries. Processes that create, change or delete orders seem to stop. When you create, change and delete orders, indicators from relational tables are deleted in the liveCache. The corresponding records stay locked until the end of the transaction. The maximum number of locks on relational records is determined with parameter MAXLOCKS. If a transaction (i.e. the record locks of one user) requires more than 10% of the lock list for locks onto a table, all the various line locks are converted into a table lock (escalation). If several transactions escalate at the same time, wait situations occur since only one transaction can set the table lock. In this cases: 1) The number of record locks held for each transaction should be as small as possible. 2) Increase MAXLOCKS if in transaction st04: a) Avg. entries are close to the value for MAXLOCKS. b) Maximum entries is equal to the value for MAXLOCKS. c) Escalations > 0. 3) Too high values for MAXLOCKS cause performance loss, as search procedures on very large lists take too long. If you must increase MAXLOCKS greatly, you should process less orders for each transaction on the application server. Large lock lists can impair the performance because of longer search runs within the lists. You can determine the number of the current lock entries as follows: #> dbmcli -d <lcname> -n <lcserver> -u control,control -uSQL sapr3,sap sql_execute "select count(*) from lock_holder"

Parameter	Description	
MAXRGN_REQUEST	Version	7.2.5.7
	Section	Extended
	Value	300 for MAXCPU=1, 3000 for MAXCPU>1 Tru64 3000
	Notes	433115
	Description	Defines how often a long-running liveCache session passes on the control temporarily to another executable liveCache session in the same thread (multitask operation). The smaller you configure MAXRGN_REQUEST, the more often terminates a long-running command in favor of other commands in the same thread. As a unit accesses to liveCache-internal functions are used.
MAXUSERTASKS	Version	7.2.5.7
	Section	General
	Value	Minimum: (Number of APO work processes)*2 + 4. Tru64 50
	Notes	433115, 205220
	Description	Maximum number of liveCache connections. Maximum number of simultaneously active users (database sessions). Sometimes you cannot connect to the liveCache because there is no free database session available on the liveCache. If the error –752 arose, you have to increment the value. If the liveCache is being used in the APO environment, the MAXUSERTASKS parameter should at least be set in such a way that each R/3 work process can build two connections for the liveCache. In addition, some user tasks should still be kept on hold for liveCache tools.
OMS_HEAP_COUNT	Version	7.2.5.7
	Section	Extended
	Value	1 Tru64 1
	Notes	424886
	Description	Defines the number of sub heaps of the OMS heap. In case of a single OMS heap becomes a bottleneck in multi CPU environments, multiple sub heaps may increase speed and scalability.

Parameter	Description	
OMS_HEAP_LIMIT	Version	7.2.5.7
	Section	Extended
	Value	=0 KB (infinite)
		75 % RAM for a dedicated liveCache server.
	Notes	433115, 337445, 424886, 389000
	Description	The configuration of the liveCache heap depends on the current or planned data volume (sizing).
		The liveCache process can allocate any amount of memory, when this memory will be write, the system will check if there is enough memory or not to assign new pages (e.g. in the swap area). If there is no more memory, the system will stop the process that uses the most memory (mostly the liveCache).
		The liveCache heap is a storage area that the liveCache dynamically requests from the operating system, but does not release again. This heap area includes, among other things, the local copies of OMS objects.
		The maximum requested and currently used size of the heap used by the liveCache is stored in system table OMS_MEMORY_USAGE.
		It is possible to determine the heap usage via the ABAP /SAPAPO/OM_LC_MEM_MONITOR.
		There must be enough swap space; SAP recommends 3xRAM for the swap space. The test aims to 4xRAM the space needed.
		Check the OS limits settled for the user about memory assignment and data segment.
OMS_HEAP_THRESHOLD	Version	7.2.5.7
	Section	Extended
	Value	100 % Tru64 100
	Notes	433115, 419634
	Description	Percentage of the liveCache heap, which if exceeded causes transactional simulations to be paged out into the data cache.
		Determines a percentage rate. If the percentage rate is exceeded when the available heap is occupied, then objects that were read and not changed within a version are removed from the heap at the end of the transaction.
		Where memory bottlenecks are concerned, it might be wise to determine a smaller value.

Parameter	Description	
OMS_VERS_THRESHOLD	Version	7.2.5.7
	Section	Extended
	Value	2097152 KB (2 GB) Tru64 100
	Notes	433115, 419634
	Description	Defines the size of an OMS version (in KB) when the liveCache starts to reduce the heap usage of those versions, i.e. at transaction end all unchanged objects are removed from the version context.

At the end of the transaction, the system deletes unchanged data in the heap from versions of a session if the version occupies more than OMS_VERS… KB of memory.

If the stored object is accessed again at a later stage within the version, the object must be copied again from the data cache into the heap. You do not have to do this if you set the OMS_VERS_THRESHOLD higher and there is enough memory available. |

Parameter	Description	
	Version	7.2.5.7
	Section	Extended
	Value	3600 seconds Tru64 3600
	Notes	433115, 325384
REQUEST_TIMEOUT	Description	Timeout in seconds while waiting for an exclusive SQL lock.

Checkpoints in the liveCache can cause wait situations. If after REQUEST_TIMEOUT seconds, not all current transactions are closed; the checkpoint terminates with the message "500 Lock request timeout".

Checkpoints should not terminate regularly. There are several reasons for this:

a) If checkpoints terminate, very long waiting times already occurred which might not be accepted by the users.

b) Between two online backups (version 7.2), a checkpoint must be written.

c) Longer periods between checkpoints increase the time needed for the reproduction on the basis of the logs in the case of a liveCache recovery.

If checkpoints nevertheless terminate regularly, determine which APO transactions run at the corresponding time and thus prevent the checkpoint. You must determine with dbmcli which work processes are holding open transactions in the liveCache and identify the current COM routine of the task.

Maximum waiting time until a requested checkpoint terminates if running transactions do not release locks.

How long seconds the checkpoint may wait until the current write transactions are closed.

A checkpoint is used to write the data pages that were changed in the Data Cache to the data DevSpaces. The checkpoint ensures the transactional consistency of the liveCache.

If a checkpoint is requested in the liveCache, the checkpoint must wait until all current write transactions are closed (Commit/rollback). During this time, new write transactions must wait. The new write transactions can start immediately after all current write transactions have been closed; they do not have to wait until the data was written to the DevSpaces. |

Parameter	Description	
SESSION_TIMEOUT	Version	7.2.5.7
	Section	Extended
	Value	0　　　　　　　Tru64 0
	Notes	214112, 325384
	Description	SESSION_TIMEOUT must be higher than LOCK_TIMEOUT or must be set with the value 0. COM routines act as write transactions in liveCache. If the Transaction does not run in liveCache for a long time, a Lock Inactivity Timeout occurs. If the session calls a COM routine later, the transaction in the liveCache is rolled back. For long running transactions (for example creating a planning board) in the APO environment, LiveCache errors (for example timeout object not found -4016) occur during a commit statement. System log (trans sm21): Deadlock occurred Database error -60 for COM Work rolled back,LOCK INACTIVI Function COMMIT on connection LCA failed Perform rollback Run-time error "DBIF_DSQL2_NATIVE_CR_ERROR" occurred File knldiag (error messages in liveCache log): "omsCommit : DbpError N" where N=-9999, -28006, -28814
TRACE_PAGES_GC	Version	7.2.5.7
	Section	Extended
	Value	20　　　　　　Tru64 0
	Notes	424886
	Description	Size (in pages) of the memory used for kernel trace of task(s) of type GC.
	Version	7.2.5.7
	Section	
	Value	
	Notes	
	Description	

LiveCache & DBMCLI Commands

The liveCache command line interface is a good tool for performing special tasks and for checking for information distributed among different DB tables in the system.

- **Which work processes are holding open transactions in the liveCache:**

  dbmcli -d <lcname> -u control,control –uSQL sapr3,<passwd> sql_execute "select appl_process, appl_nodeid, lockmode, process from sysmon_task_detail s, transactions t where s.dbpid = t.process and lower(t.lockmode) = 'exclusive'"

- **Identify the current COM routine of a task**

  dbmcli -d <lcname> -u control,control show t_c t<TaskID>
  where TaskID is the number for process in the dbmcli command above.

- **Activate the detailed time measurement of execution and I/O times Note 444602**

  You want to estimate more accurately what influence the physical I/O has on the execution times of the COM routines in the liveCache.

  The following parameters must be specified in the transaction SM49 in the selection 'dbmcli':

    -u control,control -n <LC host> -d <LC name> db_cons time enable

  From the time of the activation of the time measurement the runtimes are logged for every link between the APO work process and the liveCache.

  With SQL Studio:

  SELECT dbpid, taskname, (CMDCNT * AVGCMDEXECUTETIME) ProcessingTime, (SIOREADCNT * decode (SIOREADAVGABSTIME, 0, 0, SIOREADAVGABSTIME)) + (SIOWRITECNT * decode (SIOWRITEAVGABSTIME, 0, 0, SIOWRITEAVGABSTIME)) + (DIOREADCNT * decode (DIOREADAVGABSTIME, 0, 0, DIOREADAVGABSTIME)) + (DIOWRITECNT
  * decode (DIOWRITEAVGABSTIME, 0, 0, DIOWRITEAVGABSTIME)) IoTime
  FROM sysmon_task_detail WHERE taskname = 'user'

  For every liveCache user task (TaskID), that is every link between APO and the liveCache, the accumulated execution time in the liveCache (ProcessingTime) is displayed, as well as the I/O time contained in it (IoTime). If necessary, it is possible to assign the liveCache user task to an APO work process via the liveCache monitor (transaction LC10->liveCache Console->Process Overview).

- **Update statistics for tables in liveCache** Note 389770

- **Memory already occupied by the liveCache core process** Note 337445

  You have to check the parameters below in liveCache, using:

  dbmcli -d <LC name> -n <LC server> -u control, control param_directget

  Data Cache (8KB) = DATA_CACHE

  Converter Cache (8KB) = CONVERTER_CACHE

  Storage (KB) = dbmcli -d <LC name> -n <LC server> -& control, control show storage

Task Stack (KB) = MAXUSERTASKS * _MAXTASK_STACK

Total = DATA_CACHE * 8 + CONVERTER_CACHE * 8 + Show Storage + Task Stack + 100 * 1024 KB.

- **Determine the number of the current locks entries**

  dbmcli -d <lcname> -n <lcserver> -u control,control -uSQL sapr3,sap sql_execute "select count(*) from lock_holder"

  dbmcli -d <lcname> -n <lcserver> -u control,control -uSQL sapr3,sap sql_execute "select * from lockliststatistics"

  dbmcli -d <lcname> -n <lcserver> -u control,control -uSQL sapr3,sap sql_execute "select * from lockstatistics"

- **Start garbage collector even if the Data Cache is not 80% full or the DevSpaces 90% Note 441794**

  dbmcli -n <lcserver> -d <lcname> -u control,control -uSQL sapr3,sap -sql_execute delete object storage

- **Number of history pages in the liveCache Note 443149**

  1. Number of occupied pages OcP: APO lc10 → liveCache Performance → DevspaceStatistics → Used perm data space → Pages

  2. Number of container pages CoP: APO lc10 → liveCache Monitor → Class container tab → total sum of the column Container size

  3. Number of pages in relational tables RtP: dbmcli –d <lcname> -u control,control -uSQL sapr3,sap sql_execute "select sum(numeric_value) from pages where owner = 'SAPR3'"

liveCache Tables

LiveCache tables are described in this section, as well as the OSS Notes where these tables are referenced and a description.

	Notes	Column	Description
liveCache			
OMS_MEMORY_US AGE /SAPAPO/OM_LC_ MEM_MONITOR	389342 389000	TOTAL_HEAP_USAGE	This is the current heap usage of all COM routines, including the internal program structures of the COM routines and the OMS objects themselves. The difference TOTAL_HEAP_USAGE – USED_BY_OMS_MALLOC describes the approximate memory reserved by OMS objects in the liveCache heap.
		RESERVED	This is the amount of main memory allocated that is not returned to the OS until liveCache stops. The value specifies the maximum that has been allocated by the liveCache until that time for local copies of OMS objects and through the internal structures of APO COM routines. The limit for this value is specified with the liveCache parameter OMS_HEAP_LIMIT.
		USED_BY_OMS_MALLOC	This is the memory requested by the COM routines (components of the libraries SAPAPO.dll, SAPATP.dll for example) for internal program structures. The OMS objects stored in the heap are not contained in this.
/SAPAPO/ORDMAP	389770		
/SAPAPO/ORDKEY	389770		

APO System Configuration

Memory requirement [Note 392852]

To avoid a swapping of the system that causes performance bottlenecks, the total memory required by all applications including liveCache should not exceed the size of the physical memory or the memory managed by the operating system.

Transactional Simulation and Tru64

LiveCache version 7.2.5 (build 7 and 11)

With transactional simulation, liveCache allocates a lot of heap memory (see parameter OMS_HEAP_LIMIT and table OMS_MEMORY_USAGE). You can find out how much memory will be allocated depending on the number of Purchase Orders POs packages. This information comes from the report /SAPAPO/OM_PERFORMANCE in an initialized liveCache and can be helpful for a production system.

The amount of heap memory allocated also depends on the number of auxiliary heaps or sub-heaps configured (see parameter OMS_HEAP_COUNT). If this value is greater than 1, then the memory allocated will increase significantly.

When OMS_HEAP_COUNT=1, the heap memory (HM) allocated will be approximately:

$$HM \ (MB) = 6.2 \times PO$$

For instance, if the amount of PO configured in the report above is 1000 x 1000 x 500, then the OMS memory heap allocated will be 6.2 x 500 = 3100 MB. But if more sub-heaps are configured, say 4, then the factor increases by nearly ten times: HM (MB) = 60 x PO = 60 x 500 = 30000 MB.

The Data Cache DC memory allocated is not dependent on the number of sub-heaps and is approximately:

$$DC \ (MB) = 4.7 \times PO$$

In the example above: DC (MB) = 4.7 x PO = 4.7 x 500 = 2350 MB.

Then the memory heap allocated when you run a test with (N / 4.7) = PO packages (1000 x 1000 x PO), in a liveCache with a Data Cache = N MB will be approximately OMS_HEAP_LIMIT = (6.2 / 4.7) * N MB