Summary

Symptom
You intend to use the SAP In-Memory Database (SAP HANA) and need information about how to size it.

Other terms
Sizing, SAP In-Memory Database, SAP HANA, SAP In-Memory Computing

Reason and Prerequisites

o   SAP In-Memory Appliance (SAP HANA)

- SAP HANA is a flexible, data-source-agnostic appliance that allows customers to analyze large volumes of SAP ERP data in real-time, avoiding the need to materialize transformations.

- SAP HANA is a hardware and software combination that integrates a number of SAP components including the SAP In-Memory Database, Sybase Replication technology and SAP LT (Landscape Transformation) Replicator.

- SAP HANA is delivered as an optimized appliance in conjunction with leading SAP hardware partners.

o   SAP In-Memory Database

- The SAP In-Memory Database is a hybrid in-memory database that combines row-based, column-based, and object-based database technology. It is optimized to exploit parallel processing capabilities of modern multi core/CPU architectures. With this architecture, SAP applications can benefit from current hardware technologies

- The SAP In-Memory Database is at the heart of SAP offerings like SAP HANA that help customers to improve their operational efficiency, agility, and flexibility.

Solution
Sizing SAP HANA In-Memory Database

Important Note: SAP HANA In-Memory Database can now be sized using SAP QuickSizer. Please go to http://service.sap.com/quicksizer for further information.

Please note that this section describes the sizing of the SAP HANA In-Memory Database as it is used e.g. for replication of ERP data coming from an SAP ERP system. Other applications running on top of SAP HANA may have application specific sizing algotithms.

A comprehensive overview of the key sizing guidelines can be found in the attached slide deck "SAP_HANA_In-Memory_DB_Sizing".

o   Memory Sizing

- How to determine the Database Footprint of the data

Memory sizing of HANA is determined by the amount of data that is to be stored in memory, i.e. the amount of disk space covered by the corresponding database tables, excluding their associated indexes. Note that if the database supports compression, the space of the uncompressed data is needed. Based on this amount of data, we apply a compression factor to determine the size of the RAM needed for HANA

If data comes from an SAP NetWeaver based system, the attached shell script "get_size.csh" for DB6 and Oracle based NetWeaver systems) or "get_db4_nonbw_size.sh" (for DB4 / iOS based systems - please read the attached README file) can be used to obtain the database footprint of the tables (for DB6 and Oracle based NetWeaver systems).

The scripts can be used in two different modes:

- without parameter: the names of all active tables of the NetWeaver system along with their database size (in MB) will be written to a file "TableSize.txt" which also contains the sum of all table sizes.

- with a filename as parameter: the file should contain a list of tables that are to be replicated (one table name per row, no blanks). Only for those tables which can be found both in the file and in the database (as active tables of the NetWeaver system) the script writes the table names and their respective databse sizes to the file "TableSize.txt".

If data comes from non-NetWeaver systems or out of other database platforms, please proceed as follows:

DB2 for LUW (DB6):

The amount can be estimated using the system view "syscat.tables", assuming hat database statistics are up to date. For each table multiply "npages" by pagesize (and by "avgrowcompressionratio" if deep compression is used).

DB2 for IBM i (DB4):

The amount can be estimated using the catalog view "systablestat" in library "qsys2". For each table use the following expression to calculate the size in megabytes:

((number_rows * data_size) /

  (CASE (number_rows + number_deleted_rows)

      WHEN 0 THEN 1

      ELSE (number_rows + number_deleted_rows)

END)) / 1024. / 1024.

MaxDB:

The amount can be estimated using the system table
"tablesizes". Note that the size contains the primary key,
however, the impact on sizing is only moderate.

MS SQL Server:

Use stored procedure "sp_spaceused 'table name'" for each table
individually.

Oracle:

Use the "bytes" field in system view "dba_segments" (or other
*_segments views) for segment types "Table" and "Table
Partition". As an alternative, you can also multiply "num_rows"
by "avr_row_len" from system view "*_tables".

- Exceptional compression behaviour

In certain cases the actual compression factor can deviate from
the average compression that has been observed for various
standard scenarios. In particular, database tables which
contain data that have already been compressed on application
level usually cannot be compressed further. Data distribution
can also have significant impact. Columns with a low number of
distinct values compress better than columns where nearly all
records have different values.

Data that is loaded into HANA will be initially stored as a so
called delta index which is uncompressed. This delta index will
be merged into the existing (compressed) index at a later point
in time. If a huge amount of data needs to be loaded in one
request, the amount of memory that is needed to store the delta
index can be as much as two times the size of the delta data
itself.

o CPU Sizing

CPU resources for HANA are implicitely defined by the certified
hardware platforms that satisfy the memory requirements and that
are available from your hardware vendor. To estimate the maximum
number of active users (i.e. users which cause any kind of activity
on the server within the time period of one hour) that can be
handled by a HANA server, we have chosen the following approach.

Like in the sizing algorithms for SAP BW and SAP BWA which have
been implemented in SAP QuickSizer (for details please refer to
http://service.sap.com/quicksizer) we assume that HANA queries can
be divided into three categories ("easy", "medium", "heavy"), which
differ in the amount of CPU resources that they require. Typically,
"medium" queries use twice as much resources as "easy" ones, while
"heavy" queries require ten times as much resourses.

Furthermore, we assume that HANA users can be devided into three categories ("sporadic", "normal", "expert"). The user categories are defined by the frequency of query execution and the mix of queries from different categories. "Sporadic" users typically execute one query per hour, and run 80% "easy" queries and 20% "medium" queries; "normal" users execute 11 queries per hour, and run 50% "easy" queries and 50% "medium" queries; and "expert" users execute 33 queries per hour, and run 100% "heavy" queries. Together with a default distribution on the user categories (70% sporadic, 25% normal, 5% expert) and results from multiuser load tests on certified hardware an average resource requirement of currently 0.2 cores per user has been determined.

A more detailed description of the sizing algorithm can be found in the online help of the SAP QuickSizer.

Documentation

The documentation for SAP HANA is published on:

- o   http://help.sap.com/hana

- o   https://service.sap.com/hana

Note that during the SAP HANA 1.0 rampup phase SAP HANA documentation is specifically restricted to the designated SAP HANA RTC customers only.

Software Download

The components of SAP HANA and of the SAP In-Memory Database can only be installed by certified hardware partners on validated hardware running a specific operating system. Any other system or content developed with such systems is not supported by SAP. For further information please refer to the information page of the product version. Support Package Stacks (SPS) can be downloaded and applied to appliances according to agreements with the respective hardware partner.

CSS/CSN components for SAP In-Memory Database and SAP HANA

- o   BC-HAN: SAP High-Performance Analytic Appliance (SAP HANA)

  - –   BC-HAN-MOD: SAP High-Performance Analytic Appliance Modeler

  - –   BC-HAN-LOA: Load Controler

  - –   BC-HAN-REP: Sybase Replication Server

- o   BC-DB-HDB: SAP In-Memory Computing Engine

  - –   BC-DB-HDB-DBA: Database Administration for HDB

  - –   BC-DB-HDB-INS: Installation HDB

  - –   BC-DB-HDB-PER: Database Persistence for HDB

- BC-DB-HDB-SYS: Database Interface/DBMS for HDB

- BC-DB-HDB-UPG: Upgrade HDB

- BC-DB-HDB-ENG: SAP In-Memory Computing Engine

- BC-DB-HDB-MDX: MDX Engine/Excel Client

Header Data

Release Status:	Released for Customer
Released on:	13.12.2011 15:20:02
Master Language:	English
Priority:	Recommendations/additional info
Category:	Customizing
Primary Component:	BC-DB-HDB SAP In-Memory Database

The Note is release-independent

Related Notes

Number	Short Text
1642937	SAP HANA Appliance Software SPS 03 Release Note
1560398	SAP HANA 1.0: Statistic Server In-Memory Computing Engine
1514967	SAP HANA: Central Note

Attachments

File Type	File Name	Language	Size
PDF	SAP_HANA_In-Memory_DB_Sizing_V1_2.pdf	E	394 KB
ZIP	get_size.zip	E	5 KB