PoC With Oracle 12*c* In-Memory Option in SAP ERP ECC Environment: Easy Implementation, Minimum Effort – Significant Benefits Confirmed / Tests With Declustered Tables

ORACLE DATABASE IN-MEMORY ACCELERATES SAP ERP ECC CO-PA REPORTS OVER EIGHTFOLD

A company with turnover in excess of USD 10 billion, which has been using SAP Oracle since around 2000, has implemented an extensive PoC with Oracle 12*c* In-Memory. How much effort is involved in deploying the Oracle 12*c* In-Memory (OIM) option? How easy or complex is the implementation of Oracle Database In-Memory? What concrete improvements are associated with OIM, especially for the processing of reports and queries from SAP ERP ECC CO-PA data or tables? And finally, what benefits does Oracle 12*c* Database In-Memory offer for table declustering and the processing of declustered SAP tables? These were the key questions that the PoC was designed to answer.

To sum up the most important results, it was confirmed that the processing of SAP ERP ECC CO-PA queries were greatly accelerated utilizing Oracle 12*c* In-Memory. In some cases the improvements were more than eightfold. In other words, the performance enhancements were immense. This was achieved with a small amount of effort and without any changes to the ERP ECC table structures thus maintaining high application transparency. Implementation of the Oracle 12*c* In-Memory Option proved simple, and on top of that, OIM delivers startling benefits in connection with the use of declustered SAP tables and table declustering.

Constructing a sandbox system

The PoC was performed by a long-standing Oracle SAP customer using data from the existing PROD system. The team used an up-to-date SAP ERP ECC Oracle 12*c* sandbox system constructed specifically to mimic production for test purposes. The database size of the SAP system was slightly under 10 TB (compressed with Oracle Advanced Compression) and the main focus of the evaluation was CO-PA data as well as table and query processing of relatively large SAP ERP ECC CO-PA data volumes.

The company was specifically interested in the potential optimization of CO-PA (Controlling Profitability Analysis) processing because of a desire to offer optimum support for business requirements with accelerated reports/queries.

To assist in-memory processing with the Oracle 12*c* option, the sandbox system was equipped with additional 15GB of RAM. A correspondingly larger disk space of a little more than 30 GB was also made available. The Oracle 12*c* database was configured for in-memory processing and the CO-PA transactions/tables required for the PoC were defined

Massive report acceleration

Tests were performed on various in-memory processing scenarios, including weekly, monthly, quarterly, and annual data. The data on disk was compressed with Advanced Compression. The similar In-Memory column store was also compressed showing a ratio of 2:1. A variety of SAP transactions were used and various runs were conducted.

As the company reports, “with Oracle 12*c* Database In-Memory, across-the-board acceleration was observed in CO-PA reports in the SAP ERP ECC environment. In some cases we saw an eightfold improvement compared to the situation without the In-Memory option.” The customer also emphasized that “Oracle 12*c* Database In-Memory can be implemented with minimum effort.”

But that’s not all. The long-standing SAP Oracle customer also performed tests with declustered tables on the basis of Oracle 12*c* In-Memory. The reason was that declustered SAP tables (tables that have been “disentangled”, so to speak) can be used to implement exceptionally important business simulations and analyses that were either impossible or extremely slow given the table cluster design. SAP supports the necessary table declustering (see SAP Notes 1835008 and 1892354).

OIM and the "Bloom filter"

First off, the relevant table candidates, such as KONV tables, were identified along with their respective candidate queries. Simple single table and more complex multi-table joins were captured. Measurements were taken directly from the SAP system for later comparison. Standard SAP processes were followed to decluster KONV. This took a little time but was a straightforward exercise. Tests were then conducted using Oracle IM.

"The PoC results were as surprising as they were advantageous, revealing that Oracle 12*c* In-Memory beneficially optimizes processing, with the use of a Bloom filter. We've never seen anything like that before." In some cases processing times without OIM were reduced from 30 minutes to less than two minutes or even less than 250 milliseconds with Oracle 12*c* In-Memory.

A Bloom filter is a special optimization technique first introduced by Oracle with Oracle 10 Release 2. Developed by Burton Bloom in the 1970s, it delivers enormous improvements in query processing. This technique enables Oracle to substantially reduce communication between processes during parallel join processing for the optimal processing of large sets of data.