

1974957 - Performance with HRP1001 table. General troubleshooting guide

Version	5	Type	SAP Knowledge Base Article
Language	English	Master Language	English
Release Status	Released to Customer	Category	Problem
Component	BC-BMT-OM (Organizational Management)	Released On	09.08.2018

Please find the original document at <https://launchpad.support.sap.com/#/notes/1974957>

Symptom

The infotype 1001 stores the relationships of the different Personnel Development objects, and data gets saved in the database in table HRP1001. This infotype serves as an important source of information for many complex functionalities and is generally expected to contain a big amount of data. The accesses to this table have been optimized, but still in some particular situations, issues related to the performance can appear.

The purpose of this article is to provide a collective guide (from HCM applications point of view) with some known and common causes for a performance bottleneck, with the corresponding recommendations to solve or reduce it.

Environment

Performance issues related to relationships infotype 1001 (table HRP1001)

Reproducing the Issue

You observe some of the possible situations:

- A background process is having a bad performance, maybe also running out of memory, and the initial analysis points to a problem related to relationships in HRP1001 table.
- An online process gets stuck or bad performance is observed; when checking the corresponding work process in SM50, it points to a massive use of HRP1001 table.
- The system is giving memory dumps, like TSV_TNEW_PAGE_ALLOC_FAILED or MEMORY_NO_MORE_PAGING, and seem related to HRP1001
- The system is giving SYSTEM_IMODE_TOO_LARGE or TIME_OUT dumps, also apparently related to HRP1001

Cause

There can be many different causes for a performance issue with HRP1001 relationships table. This article covers some of these possible causes together with relevant information which can help to get a better performance.

Please proceed to Resolution section below for detailed explanations for each of these points.

1. Technical settings of HRP1001
2. Evaluation paths
3. Impact of structural authorizations
4. Redundant data due to unnecessary splits in relationships
5. Issue apparently only at database level. Indexes.

6. Considerations regarding the size of HRP1001 and relation scenario
7. Change documents for infotype 1001
8. Database specific issues. Oracle statistics.
9. Further actions: tracing performance in HRP1001 and updating to last performance corrections.
10. Considerations regarding external accesses to HRP1001

Resolution

In this section, each of the described causes and points will be explained.

1. Technical settings of HRP1001

Buffering

By standard the buffering configuration of HRP1001 is "Buffering not allowed". This table is expected to have big amounts of data, and also will receive frequent data accesses and modifications, so for the applications to run correctly and with the better performance possible, the buffering needs to be off. Activating buffering for HRP1001 is not supported, and would probably cause selection errors (all the standard accesses to the table have been designed considering the condition of no buffering) or performance issues. If you are having performance or some other issues with HRP1001, ensure first that the buffering is OFF (standard default value).

Table logging

The table logging could have an impact on performance of HRP1001 updates. Sometimes it might be considered useful -for auditing or analysis purposes- to temporarily activate the table logging of HRP1001 (for example to keep track of unwanted deletion of relationships). But in general, it would be better to discard this option for the long term as it would affect the performance for each update done to HRP1001 table, because the relationships table is frequently updated in productive systems. Note [1916](#) describes more information on this regard.

2. Evaluation paths

Recursions happening with evaluation paths

This kind of problematic situations can either happen because:

A) incorrect relationships which are creating an infinite loop in the org structure, when some evaluation path is used

B) incorrect evaluation path with a definition in T778A searching objects in an infinite loop

Both notes [608767](#) and [1972083](#) help with these situations, and provide an example and hints for the resolution of these cases.

Evaluation paths navigating the structure in a very resource demanding way

Sometimes there could be customer evaluation paths which cause massive search conditions when they are triggered, so the recommendation in these cases would be to reduce the definition of the evaluation path as much as possible, or to reduce the amount of root objects selected, separating the execution in smaller independent runs.

Also performance bottlenecks might appear with some standard evaluation paths which have a very general search condition (in T778A definition), if they are used for a big amount of initial root objects. In these cases the recommendation is to reduce the selection of initial data as much as possible. The best approach is usually to execute same selection, but running it in some smaller batches of initial objects, so at the end the same amount of objects get selected, but in a more balanced way.

3. Impact of structural authorizations

One of the frequently biggest impacts on performance in HRP1001 table comes from the Structural Authorizations functionality. The reason is that the structural authorizations need to retrieve high amounts of

relationship data corresponding to the objects related to each profile. In case structural authorizations are active, with very big structural profiles, it is very convenient to regularly update the index (INDX) to get a better overall performance (process done with RHBAUS* reports). Notes with additional information on this topic can be found in PA-BC area.

Bottlenecks with structural authorizations usually begin after a call to function RH_STRU_AUTHORITY_CHECK, so identifying a call to this function (in performance traces or dumps) might help to find out if the bottleneck is really originated by the structural authorizations.

4. Redundant data due to unnecessary splits in relationships

In some customer scenarios, there could be relations in HRP1001, which are splitted into different records with subsequent time periods, while the rest of the data is exactly the same. When the relationship dates of these subsequent records (where the rest of data is the same) are next to each other, with no gaps, then these records can be represented with an unique record corresponding to the total time period interval. These types of splitted records can arise for technical reasons when integration between Personnel Administration and Organizational Management is involved. So in scenarios with many of these redundant records, the size of HRP1001 can be reduced, and depending on the amount of splitted records to be merged, this can cause a positive impact on performance.

There are two similar standard reports which can be used to merge splits in infotype 1001: RHUNIT00_BASE was introduced for systems with no HR component available (as organizational management belongs to SAP_BASIS module), and RHUNIT00 report existing in ERP systems with HR component. Proceed to note 539711 and to the reports documentation for more details. It is important to consider that these reports would have a long runtime if many objects are selected, so they might require appropriate splitting in smaller selection groups. Also please consider that RHUNIT00/RHUNIT00_BASE reports are not designed as inconsistency fixing tools, but just as merge tools for records which are originally consistent.

5. Issue apparently only at database level. Indexes.

In general it is not recommended to modify or create new indexes for HRP1001 table, as the ones existing by standard are considered the optimal ones, for all the expected standard use of the table. Having many indexes would have an impact on performance, because they will need to be updated with each change in HRP1001, so that is why in general it is not recommended to have more than the standard defined ones.

6. Considerations regarding the size of HRP1001 and relationship ratio scenario

Having a huge HRP1001 table would directly affect performance, but there are other situations which could possibly affect performance, without necessarily having a huge amount of total entries in HRP1001. An example of these situations would correspond to organizational structures where the relationship ratios are not coherent and balanced, like for example having one organizational unit which has 10.000 relationships to children positions below it, etc. Such sample 1:10.000 relation ratios are not expected in productive organizational structures, and then there is the risk that performance issues would appear with that specific parent object. It is much recommended here, if possible by business requirements and definition, that the organizational structures are then stored in a more balanced way with not such high ratios, having better balanced ratios like for example 1:100, 1:200, etc.

7. Change documents for infotype 1001

The functionality of change documents acts as a log at application level for updates in infotypes. Sometimes it might be useful or required by business to have this log active for infotype 1001. Activating this log massively could have an impact in the performance of each update on HRP1001, so, if possible, it is recommended to reduce the log of change documents to the minimum relationships required at business. The change documents are configured in view T77CDOC_CUST, and there is also some more information in

note [1158326](#) and documentation of report RHCDOC_DISPLAY.

8. Database specific issues. Oracle statistics.

The performance of HRP1001 could sometimes be affected (or improved) by configuration from the specific database technology. For example, for Oracle database systems, the statistics definition would be relevant for HRP1001 access; for more details about this, note [1020260](#) can be checked, and the Oracle expert team could be contacted.

There are other corrections and recommendations for the different database systems, like for example note [862828](#) for Microsoft SQL server databases, note [1373842](#) for DB2 database systems, etc. It is recommended to check for possible updates or standard recommendations on this regard in the notes and documentation, and also with the corresponding Basis and Database expert teams.

9. Further actions: tracing performance in HRP1001 and updating to latest performance corrections

When there is a performance issue with HRP1001 table, usually the SQL traces would show a big amount of selections to the table, but it is more important to determine first the scenario and conditions under which such massive work is triggered. Many times the issues are faced because non-released functions (see note [109533](#)) are being used in a way which was not expected in its design, therefore causing a huge selection of data which was not expected nor prevented. In these cases the solution would not be from the standard code, but from the custom application logic. To clarify all these situations, it is convenient to perform an ABAP trace, either from ST12 or SE30/SAT transactions.

Also, from time to time, some minor improvements are delivered for some general standard functions or business processes related to HRP1001 accesses. For this reason, it is convenient to search and apply latest correction notes which describe performance improvements affecting this table.

10. Considerations regarding external accesses to HRP1001

HRP1001 table serves as an important source of data for many different functionalities working with organizational management relationships. When there are performance bottlenecks, sometimes they only happen in a specific functionality (represented by a different SAP component), while the rest of the tools and processes with HRP1001 work fine. This usually means that either too much data is being selected from HRP1001 in that particular process, or that this process logic has not been fully optimized for high data volumes.

Some examples of these adjustments which were needed are PE-LSO notes [1994434](#) and [1774521](#), PA-ER note [1821076](#), etc and basically the purpose of these notes is to reduce or optimize the external retrieval of information from HRP1001. As that data recovery is based on how each functionality accesses HRP1001 table, the solutions are coming from the different areas, and not from the general organizational management component BC-BMT-OM.

These situations can be difficult to detect at first, so the best initial approach would be to determine whether a performance bottleneck in HRP1001 table is coming from a particular functionality only, or instead if it is something more general which can be observed and reproduced also from some of the common organizational management transactions such as PP01 or PPOME. In case the performance happens only in an specific area (PA-TM, PE-LSO, PA-ER, Industry solutions, etc) it is convenient to search and apply the latest notes from that area related to observed logic. Also previous sections 2 and 9 can be relevant for these cases, and usually a good approach when the involved functionality is not clear, is to perform a source code trace (ST12 or SAT) and try to determine the area where the bottleneck would be really starting.

Keywords

Performance TIME_OUT MEMORY_NO_MORE_PAGING TSV_TNEW_PAGE_ALLOC_FAILED
 RH_STRUC_GET OBJID HRP1001 buffering recursion SAPLRHAP UPDATE_BUFFER PPOMA_CRM

Products

SAP ERP all versions

SAP R/3 all versions

SAP enhancement package for SAP ERP all versions

SAP enhancement package for SAP ERP, version for SAP HANA all versions

Other Components

Component	Description
PA-ER	E-Recruiting
PE-LSO-TM	Training Management
PA-BC	HCM Authorization and HCM Basis
PA-PA-XX	General Parts
PA-OS	Organizational Plan

This document refers to

SAP Note/KBA	Title
1972083	How to analyze a recursion in organizational management: a practical example in PA40