

SAP Note

**3232246 - feature not supported: CONTAINS predicates on expression "<expression>"**

Version 14 Validity: 18.09.2023 - active

Language English

Header Data

Released On	18.09.2023 02:41:48 By Jinsol Kim (I533393)
Release Status	Released for Customer
Component	HAN-DB SAP HANA Database Display ACRF Content
Priority	Recommendations / Additional Info
Responsible	Jinsol Kim (I533393)
Processor	Jinsol Kim (I533393)
Category	Consulting
Relevant for Translation	No

Symptom

You run a search query with a SCORE() function and a CONTAINS() predicate with at least one calculated column using an expression as parameter instead of a pure table column:

SAP DBTech JDBC: [7]: feature not supported: CONTAINS predicates on expression <expression>

Other Terms

Fuzzy Search, Search Query, Calculated Attribute

Reason and Prerequisites

Issue number 293117

Reason:

HANA Search queries do not support calculated columns as comparison columns in *CONTAINS((col1, col2, ..., col<n>), 'term1, term2, ..., term<n>', ...)* predicates. In this context, only pure table columns or projections of such are supported.

Hence, if a calculated column is provided in the CONTAINS() predicate and this column contains any expression other than a projection of a pure table column, e.g.

- COL1+ID
- CONCAT(COL2, "ABC")
- ABAP_UPPER(COL2)
- Type casting (i.e. TO_DECIMAL, TO_INT, TO_NVARCHAR and TO_VARCHAR)

the query fails with the error message above.

This is an expected behavior.

Example Scenario

1. Create column table 'TEST_TABLE'

```
CREATE COLUMN TABLE TEST_TABLE (
  "ID" INTEGER PRIMARY KEY,
  "COL1" DECIMAL (5,2),
  "COL2" NVARCHAR(5)
)
```

2. Create calculation view 'CV' that has three attributes as follows

- "ID_CV" <= projection of column TEST_TABLE.ID

- "COL1_CV" <= calculated column expressed as COL+1
- "COL2_CV" <= calculated column expressed as CONCAT("COL2", 'TAIL')

3. Run search queries below

A. SELECT SCORE() FROM CV WHERE CONTAINS(CV.ID_CV, 100);

> no error raised because ID_CV is a pure projection of table column TEST_TABLE.ID

B. SELECT SCORE() FROM CV WHERE CONTAINS(CV.COL1_CV, 100);

> *Could not execute 'SELECT SCORE() FROM CV WHERE CONTAINS(CV.COL1_CV, 100)'*
 SAP DBTech JDBC: [7]: feature not supported: CONTAINS predicates on expression
 TO_DECIMAL(TO_DECIMAL(TEST_TABLE.COL1, 5, 2) + 1.00, 34, 2)

C. SELECT SCORE() FROM CV WHERE CONTAINS(CV.COL2_CV, 'TARGET');

> *Could not execute 'SELECT SCORE() FROM CV WHERE CONTAINS(CV.COL2_CV, 'TARGET)'*
 SAP DBTech JDBC: [7]: feature not supported: CONTAINS predicates on expression **CONCAT**(TEST_TABLE.COL2, 'TAIL')

Solution

To avoid this error, either:

1. Modify modeling of the affected calculation view so that no expression is included in any calculated columns used in CONTAINS() predicates

or
2. Materialize the data retrieved from the calculation view so that no expression is left to be processed by SQL Optimizer (please note that this approach is highly likely to result in query performance degradation with more memory usage and longer execution time)

[Materialization Example]

From the examples above in "**Reason**" section, query B and C can be rewritten as follows:

DO BEGIN

temp = SELECT * from CV WITH HINT(**NO_INLINE**);

SELECT SCORE() FROM :**temp** AS tt WHERE CONTAINS(tt.COL1_CV, 100); -- query B
 SELECT SCORE() FROM :**temp** AS tt WHERE CONTAINS(tt.COL2_CV, 'TARGET'); -- query C

END;

Please note that without HANA statement hint **NO_INLINE**, there is a possibility that materialization is skipped so this query might still fail. In case that materializing the result from the calculation view and calling it multiple times turns out to be extremely inefficient from SQL optimizer's perspective, without hint **NO_INLINE** SQL optimizer might decide to enforce the table variable :**temp** to be inlined so that the anonymous block above can be simplified internally as follows:

DO BEGIN

SELECT SCORE() FROM **CV** AS tt WHERE CONTAINS(tt.COL1_CV, 100); -- query B
 SELECT SCORE() FROM **CV** AS tt WHERE CONTAINS(tt.COL2_CV, 'TARGET'); -- query C

END;

Since this reduced form is identical to original query B and query C, the execution will fail and raise the error if SQL hint **NO_INLINE** is not used together.

Validity

This document is not restricted to a software component or software component version

References

This document is referenced by:

Customer Incident (20)

Number	Year	Title	Status	Processor	Customer Number	Customer Name
0000006488	2024	[Alert: Action Required]_PS4WKB : High number of ABAP Short	Sent to ServiceNow		0001635863	WAAK Maatwerkbedrijf vzw
0004671101	2023	Database_error_7_with_PRE_access_to_table	Restricted	Andras Letai	0000146416	Milliken & Company
0000001805		Datenbankfehler 7 beim PRE-Zugriff auf T	Restricted	Belen Ramirez	0001040097	Plaion GmbH
0000052481	2024	Database_error_7_with_PRE_access_to_table SHSM_V_PRRMP	Confirmed automatica		0000025389	Landesmesse Stuttgart GmbH
0000054049		[H]_short_dump_are_generating_in.SP1	Sent to ServiceNow		0000274825	Telstra Corporation Limited

[more...](#)

SAP Knowledge Base Articles (2)

3329686 HAN-DB [SM21 error messages : database error 7 with PRE access to table FAA_M_AANLA](#)

2380176 HAN-DB [FAQ: SAP HANA Database Trace](#)

ServiceNow Case (20)

Number	Year	Text	Status	Processor	Customer Number	Customer
0000394073	2023	Database_error_7_with PRE_access_to_table.T012K	Closed	Wallie Cheng (I819340)	0001387370	New York State Homes & Community Re
0000133900		FI-AA Migration mit der LSMW.	Closed	Bernhard Kirchner (I024452)		
0000001805		Datenbankfehler_7_beim PRE-Zugriff auf T	Closed	Belen Ramirez (I013792)		
0004671101		Database_error_7_with PRE_access_to_table.	Closed	Andras Letai (I534017)	0000146416	Milliken & Company (0000146416)
0000125616		RAISE_SHORTDUMP CX_SADL_SHORTDUMP multiple occurrences daily.	Closed	Anonymized (I575827)		

[more...](#)