Prosenjit Das Neogi

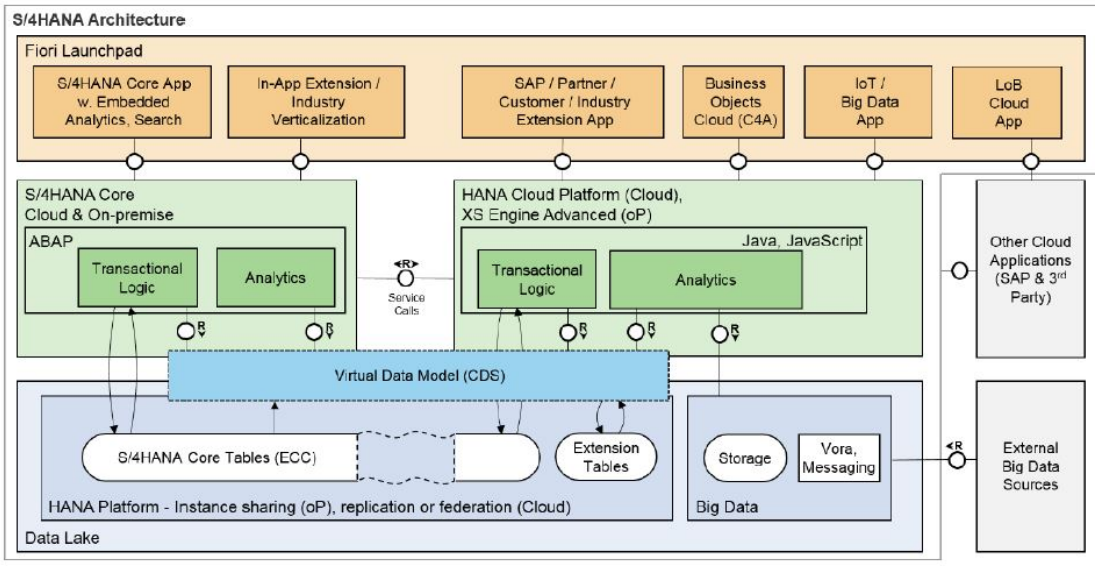
July 3, 2018 | 3 minute read

Performance Optimization for ABAP CDS view

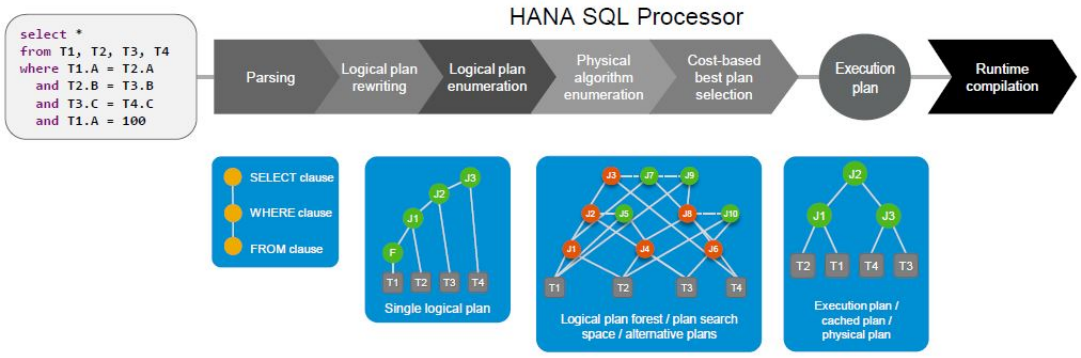
Topic : The blog will discuss on the Topic of Performance optimization for CDS views.

As an ABAP CDS view developer working in S/4HANA projects, Please find the points we need to check while developing the view.

General Architecture for SAP S/4HANA:



SQL Processing Steps :



CDS Views Complexities :

1.CDS views stacking can get so complex that it results in performance degradation

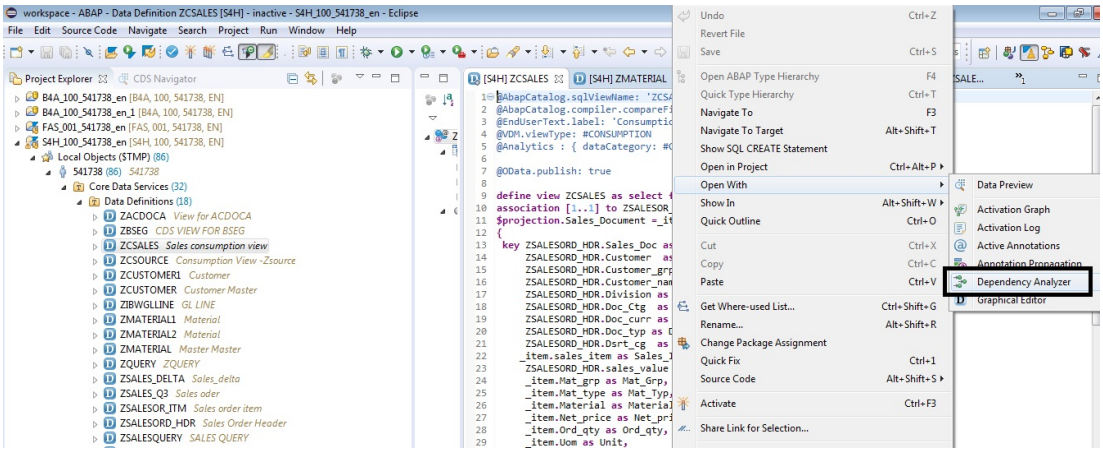
2.Number of execution plan variants raises according (#??????)?HANA optimizer could fail to find the optimal execution plan.

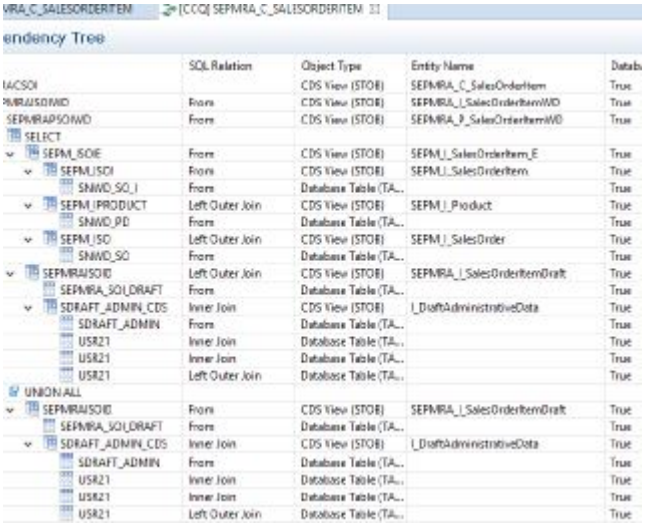
3.By using layered CDS views, formerly fast accesses to buffered tables and application caches on ABAP server might be replaced by complex database accesses

4.Declarative programming style of SQL can have a performance drawback since optimizer plans do not take into account the actual attributes of a statement. Cutting away unnecessary execution branches might not be done with the same efficiency as with imperative programming in ABAP

Display Complexity of the CDS views:

ADT (ABAP Development Tools for Eclipse) > Open with Dependency Analyzer.





Note :

Separate transactional data, master data, org data, customizing, and other metadata This is important to leverage caches on the application server for meta data.

Classification of CDS views using performance annotations :

@ObjectModel.usageType.serviceQuality
quality of service with respect to the expected performance of the CDS view

@ObjectModel.usageType.dataClass
type of data in CDS view (transactional data, master data, ...)

@ObjectModel.usageType.sizeCategory
set of data whichhas to be searched through in order to compute the result set

```
@AccessControl.authorizationCheck: #CHECK
@ObjectModel.usageType.serviceQuality: 'D'
@ObjectModel.usageType.sizeCategory: 'XL'
@ObjectModel.usageType.dataClass: 'CUSTOMIZING'
@ClientHandling.algorithm: #SESSION_VARIABLE
```

CDS views: performance annotations

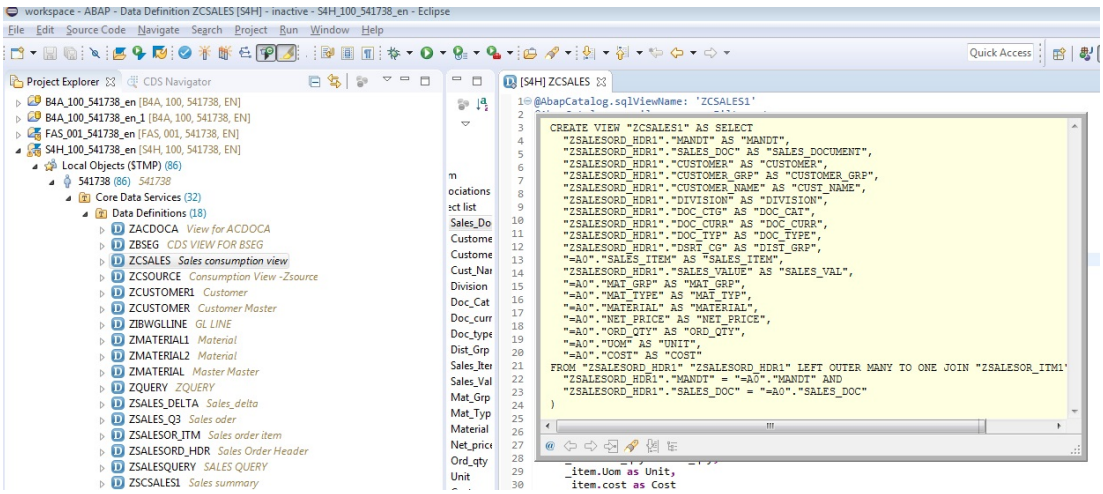
serviceQuality	Each CDS view shall be assigned to one of the following quality categories: **A**: the view may be consumed within business logic for **high volume transactions** or background processing **B**: the view may be consumed within business logic for **transactions** or background processing **C**: the view may be consumed **from the UI** in transactions for single object retrieval **D**: the view may be consumed for **analytical reporting** **X**: the view is built to **push down application code** to HANA
sizeCategory	Each CDS view shall have assigned a size category. The size category enables the consumer to judge the possible result set. It reflects the number of rows that has to be searched through to get a result. The labels correspond to the following size categories (expected number of rows in customer production systems): **S** < 1000, **M** < 100.000, **L** < 10.000.000, **XL** < 100.000.000, **XXL** > 100.000.000
dataClass	To support the decision on cache strategies for higher layers and to enable client side statement routing using these caches, each CDS view shall have assigned a data class. The different data classes correspond to different life time cycles. **TRANSACTIONAL** data is written or changed in high volume transactions **MASTER** data is read, but not written or changed in high volume transactions **ORGANIZATIONAL** data describes the organizational structure of a company and its business processes **CUSTOMIZING** data describes how a concrete business process is executed at the customer **META** data specifies how the system is configured or describes the technical structure of entities **MIXED** data shall be chosen if the CDS-View contains tables with several different of the above types

service-Quality of CDS views: requirements and KPIs:

service Quality	Usage	size Category	Expected dataClass	Number of tables	Functions	Aggregation	Data Classes	Buffering	Testing	SELECT fld FROM view WHERE key LIMIT 1	SELECT * FROM view WHERE key LIMIT 1
A	may be consumed within business logic for **high volume transactions** or background processing	S, M, L, XL or XXL	transactional, master, customizing, org, meta, **but not mixed**	**≤ 3**	no	no	all identical	buffered when all underlying tables buffered	Automatic generic	**< 1ms**	**< 2ms**
B	may be consumed within business logic for **transactions** or background processing			**≤ 5**	only if applied to result set (conversions)					**< 2ms**	**< 5ms**
C	may be consumed **from the UI** in transactions for **single object retrieval**, may not be used within application logic.		mixed	**≤ 15**		no aggregation of a huge number of table rows				**< 10ms**	**< 20ms**
D	may be consumed for **analytical reporting**	XXL	mixed	**< 100**					Manual	define realistic test case on realistic test data and define KPIs (default < 500ms)	
X	is built to **push down application code** to HANA									As class D **or improved performance and throughput compared to implementation without code push-down**	

Generate SQL statement for CDS Views:

ADT (ABAP Development Tools for Eclipse)> View SQL statement.

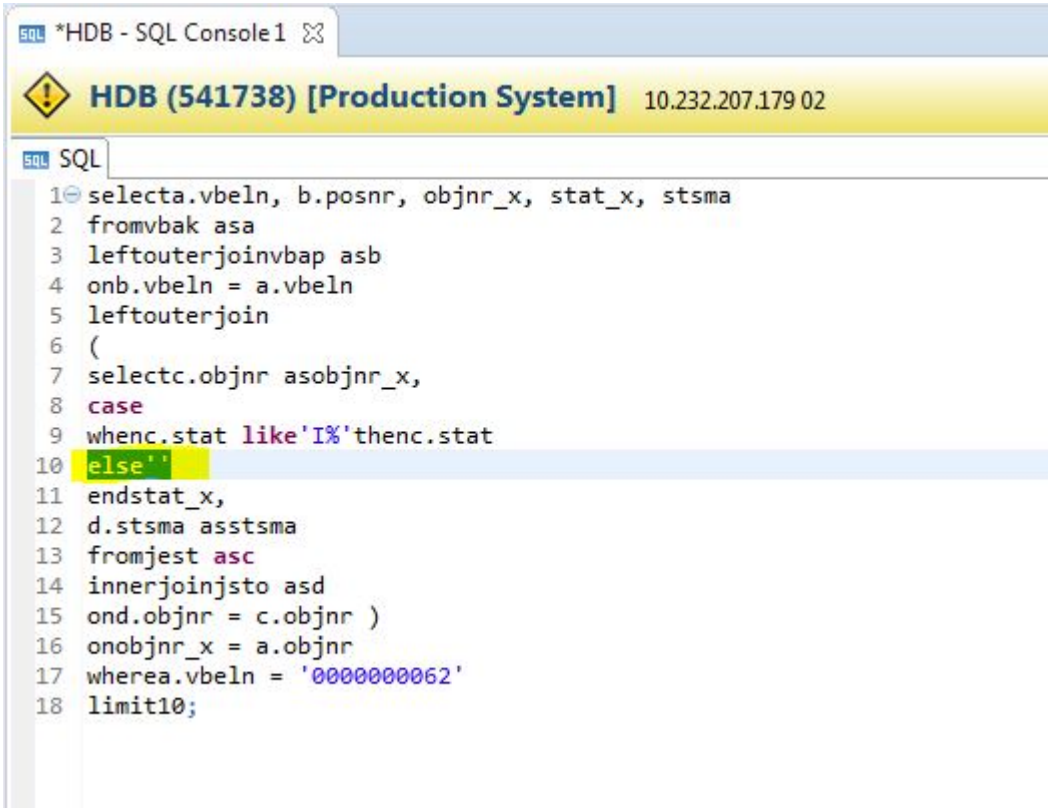


SQL Tips :

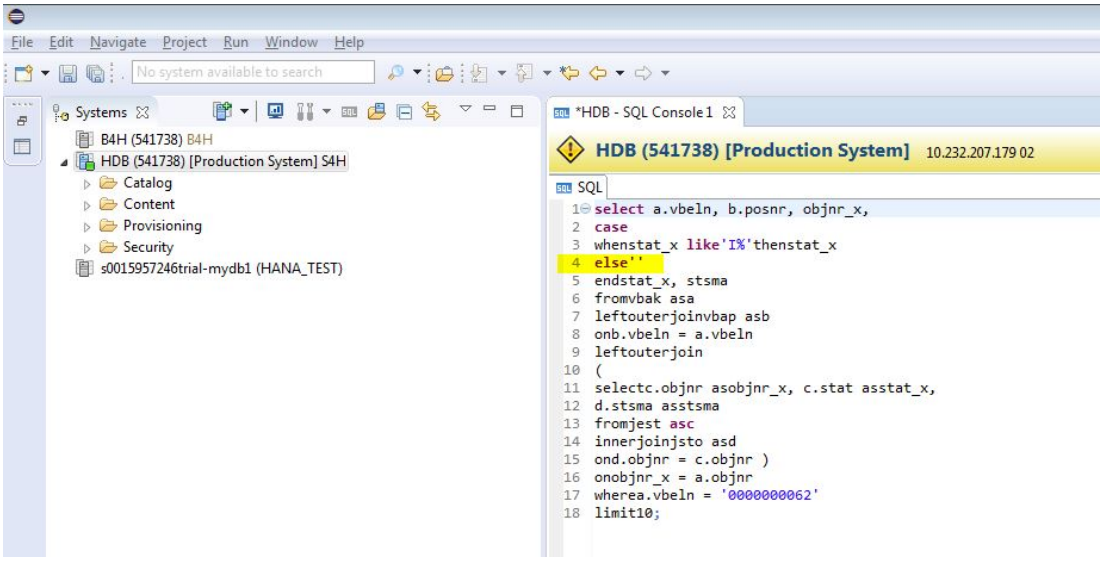
Exposed non NULL preserving calculated field

The blow example 2. Result 385% improvement of Query execution :

Example 1: The output time was 883 ms



Example 2: The output time was 34 ms



Limit push down and aggregation push down :

•Order by: first sort, then limit problem for calculated fields
•Distinct operator, any aggregation
•cardinality changing joins (e.g. LoJoin TO MANY)

Rounding (and other functions) stop optimizer from pushing down aggregations
Currency conversion and all arithmetic operations except +/-contain implicit rounding functions

Special Note on Performance Recommendation:-

1. Keep CDS views simple (in particular service-Quality A and B = #BASIC views)
2. Amount of data persisted in S4 CDS views should not exceed 20% of the overall data volume of the system.
3. In transactional processing, only use simple CDS views accessed via CDS key
4. Expose only required fields –define associations to reach additional fields when requested
5. Perform expensive operations (e.g. calculated fields) after data reduction (filtering, aggregation)
6. Avoid joins and filters on calculated fields
7. Test performance of CDS views. Test with reasonable (= realistic) test data
8. Analyze accesses to more complex views with HANA PlanViz to see whether filters are pushed down to all brances.
9. Stay tuned on caching possibilities of SAP HANA and Fiori apps

Alert Moderator

Assigned tags

SAP S/4HANA

SAP Fiori for SAP S/4HANA

embedded analytics

Similar Blog Posts

CDS view performance Annotation for CDS view performance

By Abhimanyu Sharma Feb 06, 2020

ABAP CDS view: distance in working days between two dates

By Sergey Shablykin Feb 06, 2018

Easy way to prepare dataset for performance test of ABAP-CDS-views in S/4

By Sergey Shablykin Oct 03, 2017

Related Questions

SAP ABAP CDS

By Former Member Dec 24, 2015

Best Practices For ABAP CDS VDM development and front-end development

By Pierfrancesco La Spada Oct 04, 2017

ABAP CDS View - Sequential read - Performance Issue

By Arthur Alexis Jul 29, 2019

5 Comments

You must be Logged on to comment or reply to a post.

Vivek Tripathi
July 3, 2018 at 7:04 am

Well explained Prosenjit....

Like 1 | Share

Vikash C Agrawal
July 10, 2018 at 8:44 am

Very well thought out blog. Thanks.

Like 1 | Share

Prosenjit Das Neogi | Blog Post Author
July 10, 2018 at 7:41 pm

Thanks Sir

Like 0 | Share

Randolf Eilenberger
January 18, 2019 at 7:50 am

Hi Prosenjit,

Your blog seems to have been influenced by session S4H300 – Boost Performance for CDS Views on SAP HANA at TechEd 2017 (View the presentation (PDF)). Currently I publish a series of blog posts Safeguard Performance of ABAP CDS Views, which might also be interesting for you and your readers.

Like 2 | Share

Martin Sommer
March 22, 2021 at 7:25 am

Hi Randolf, can you please repair the link to the presentation PDF? Best regards

Like 0 | Share

Find us on

Privacy	Terms of Use
Legal Disclosure	Copyright
Trademark	Cookie Preferences
Newsletter	Support