Technical Articles

Todd Witter

December 12, 2019 | 2 minute read

Optimize CDS views using dbHints annotation

Discover CDS views with potential Quick Wins

Gateway performance analysis via /IWFND/STATS gives powerful tools to analyze load on backend S4 system from a volume perspective, among others. What about expensive CDS statements within S4? From SQL Editor in S4 via DBACOCKPIT -> Diagnostics -> SQL Editor we can determine which CDS views can be optimized by running following statement:

- SELECT TOP 50 plan_size_count AS plansize, *
  FROM m_sql_plan_cache
  WHERE statement_string like '%Z<CDS View(s)>%'
  ORDER BY plansize DESC

Add dbHint annotation and confirm performance benefit

Hints placed on CDS view itself @AbapCatalog.dbHints: [{dbSystem: #HDB, hint: '<hint_name>'}]. Since CDS views are not actual tables, and indices less relevant, the join engines should be considered to optimize performance: OLAP, JOIN, HEX, or ESX engines (see SAP HANA Performance Developer Guide). With CDS view(s) having potential for performance benefit, run following SQL statement, with and without hints, selecting fields causing joins or emulating long run times with searching on non-key fields, for example:

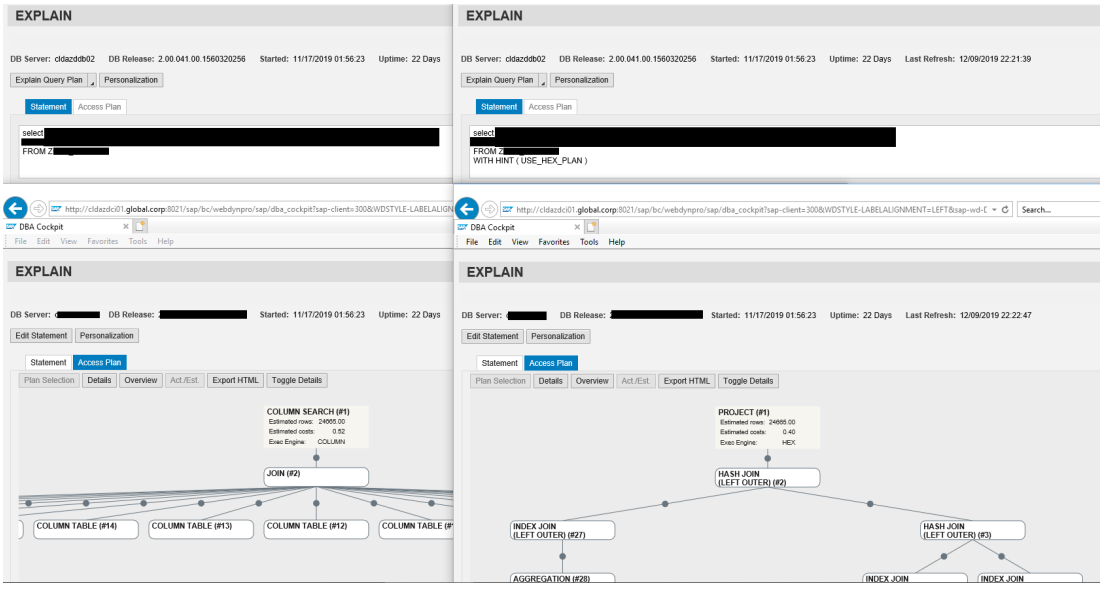
- SELECT *
  FROM Z<CDS View>
  WITH HINT (USE_ESX_PLAN)

Also try following hints, among others from relevant documentation from SAP, to understand engines and associated costs (see corresponding engines and hints below).

- Column
  - USE_OLAP_PLAN
  - NO_USE_OLAP_PLAN
  - NO_USE_HEX_PLAN
  - NO_USE_ESX_PLAN
- HEX
  - USE_HEX_PLAN
- ESX
  - USE_ESX_PLAN

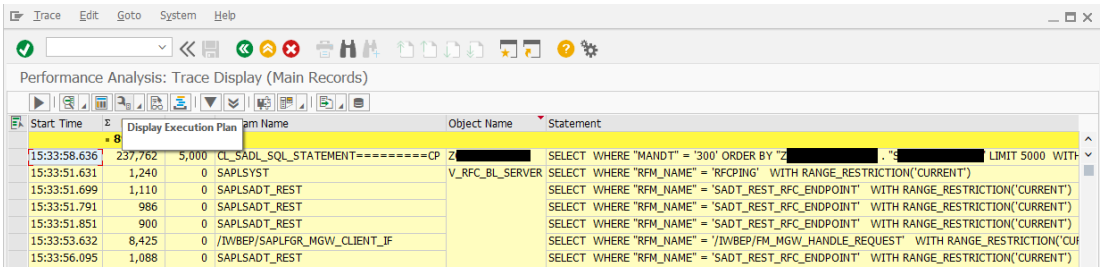
SAP HANA Execution Engine (HEX)

HEX engine is preferable in this case, with all things remaining the same.



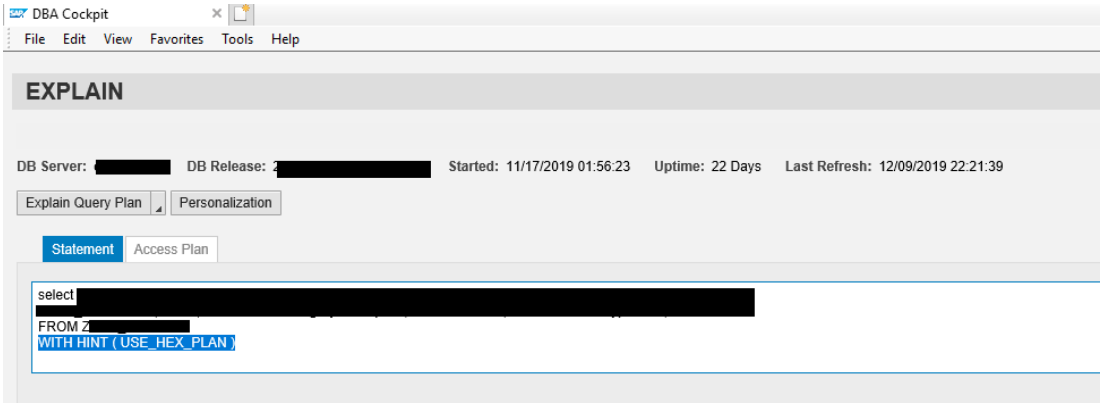
Discover current join engine

What does PlanViz show without dbHints annotation on this particular CDS view? Column, which is the same as without the hint declared in SQL Statement. To confirm set SQL trace using ST05, run odata request, display trace, Execution Trace download as .plv file (see Display Execution Plan below), and upload .plv in Eclipse. Alternatively Graphical Explain will show similar PlanViz information.

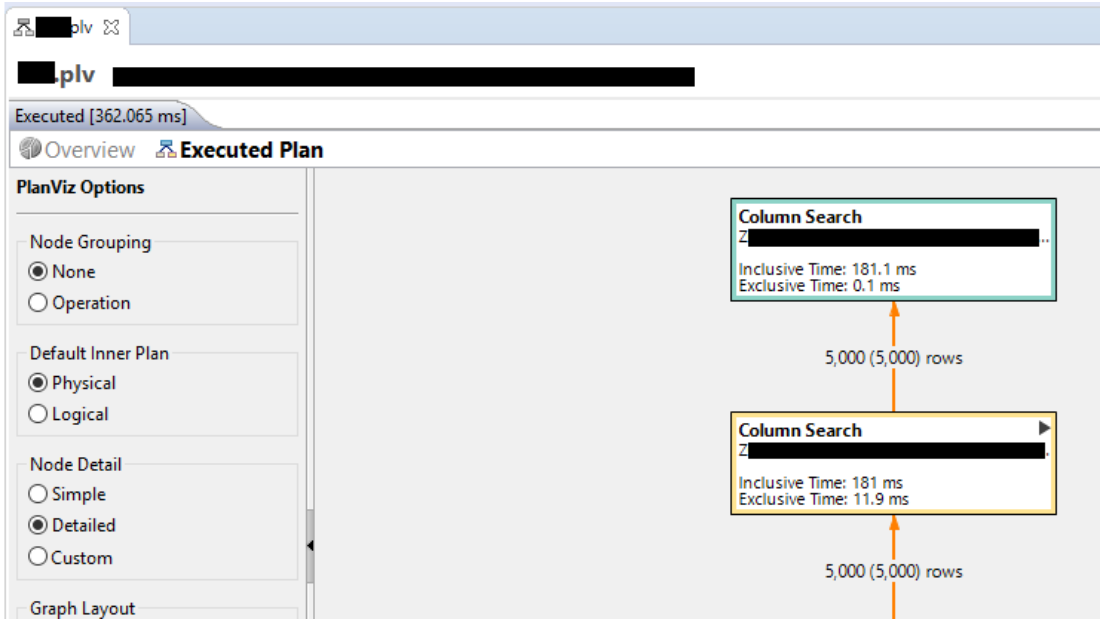


Explain Plan -> Graphical Explain will launch Browser

Run Explain Query plan



Execution Trace or PlanViz from Eclipse using Open File... -> .plv



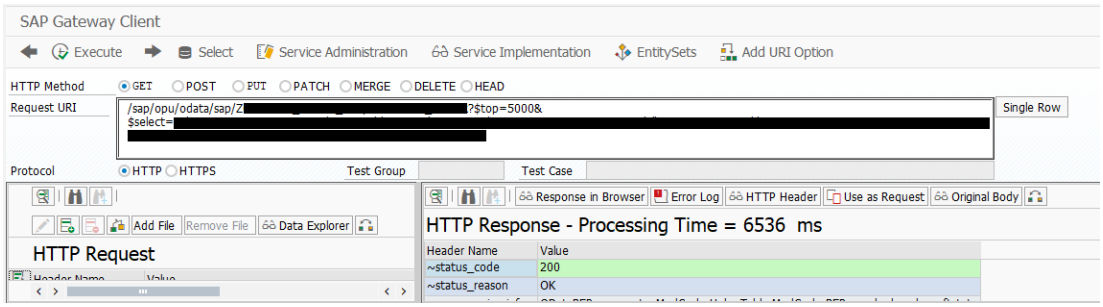
Results

Add dbHints annotation with HEX to CDS view and check performance improved:

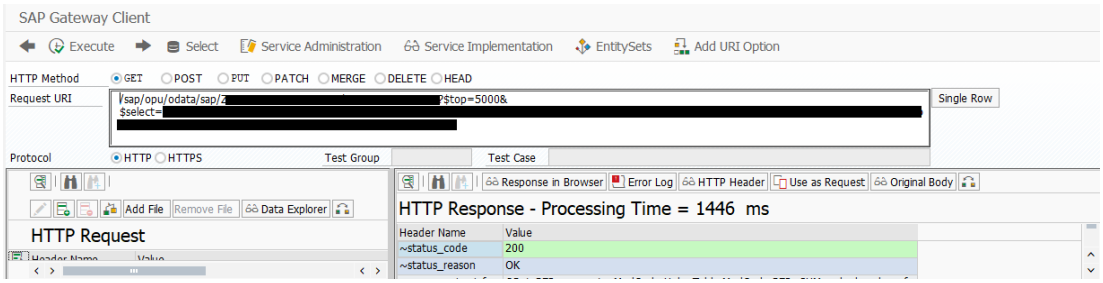
@AbapCatalog.dbHints: [{dbSystem: #HDB, hint: 'USE_HEX_PLAN'}]

Results

Odata Gateway result without hint is 6.5 seconds



Odata Gateway result with hint is 1.4 seconds



Conclusion

In conclusion the HEX engine capitalizes on SAP HANA, see note 2570371 – FAQ: SAP HANA Execution Engine (HEX). Without dbHints set to USE_HEX_PLAN in the CDS views, consumption via gateway odata requests using SADL framework, are not using the latest, most innovative query processing engine from SAP.
Thanks for reading

Alert Moderator

Assigned tags

SAP S/4HANA

SAP Analytics Cloud

Similar Blog Posts

CDS view performance Annotation for CDS view performance

By Abhimanyu Sharma Feb 06, 2020

Part#1. SAP CDS views Demystification

By Sanjeev Kumar Oct 21, 2019

Part 1 - SAP S/4HANA Cloud Data Integration to power your Intelligence Enterprise

By Michael Sanjongco Apr 19, 2021

Related Questions

Issue with "Create KPI" (Fiori App) on CDS Views (S/4HANA 1511)

By Former Member Feb 12, 2016

How to reference analytical CDS query view to a SEGW OData project data model ?

By Momen Allouh Aug 16, 2021

How to work with CDS Views that are not released?

By Ninette Krebs Feb 21, 2018

Be the first to leave a comment

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

Find us on

Privacy	Terms of Use
Legal Disclosure	Copyright

Trademark	Cookie Preferences
Newsletter	Support