

ABAP Core Data Services

SAP Business Suite – Best Practice Guide

PUBLIC



Document Version: **February 7, 2020**

www.sap.com/contactsap

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are





TABLE OF CONTENTS

CONTRIBUTORS	4
Authors	4
Acknowledgements	4
Feedback Welcome	4
INTRODUCTION	5
What's new about ABAP Core Data Services	5
About this Guide	6
ABAP CORE DATA SERVICES	8
Availability and Core Capabilities	8
CDS features vs. SAP Release	8
Documentation Links	9
Development Framework	9
Services Based on CDS Views	10
SAP Applications using CDS	12
BEST PRACTICES	14
SAP Applications	14
Custom Applications	14
Development Rules and Governance	15
Performance Safeguarding	16
DATABASE SPECIFICS	17
General Remarks	17
IBM Db2 for i	19
IBM Db2 for Linux, Unix and Windows	22
IBM Db2 for z/OS	28
Microsoft SQL Server	30
Oracle Database	32
SAP Adaptive Server Enterprise (SAP ASE)	36
SAP HANA DB	37
SAP MaxDB	38
TRAINING	39
APPENDIX	40
List of changes	40
Figures	40
References	41

CONTRIBUTORS

Authors

Task / expertise	Name	Mail address
Coordination General content	Bernd Kohler	bernd.kohler@sap.com
SAP NetWeaver ABAP	Gaspar Zoltan Erdelyi	gaspar.zoltan.erdelyi@sap.com
Performance	Randolf Eilenberger	randolf.eilenberger@sap.com
IBM Db2 for i	Dorothea Stein	dorothea.stein@sap.com
	Christian Bartels	christian.bartels@de.ibm.com
IBM Db2 for Linux, Unix and Windows	Thomas Rech	thomas.rech@de.ibm.com
	Dirk Nakott	nakott@de.ibm.com
	Karl Fleckenstein	karl.fleckenstein@de.ibm.com
IBM Db2 for z/OS	Bernd Kohler	bernd.kohler@sap.com
Microsoft SQL Server	Tino Rosenkranz	tino.rosenkranz@sap.com
Oracle Database	Matthias Lienert	
	Patrick Kolb	patrick.kolb@sap.com
SAP Adaptive Server Enterprise	Andreas Fischbach	andreas.fischbach@sap.com
SAP HANA DB	Irena Kofman	irena.kofman@sap.com
SAP MaxDB	Thomas Elvers	
	Torsten Pfeiffer	

Acknowledgements

The authors would like to thank the following SAP colleagues for their valuable contributions:

- Timm Falter
- Frank Hoffmann
- Jörg Maier
- Carine Tchoutouo Djomo

Feedback Welcome

ABAP Core Data Services (CDS) is still a rather new technology. Therefore, this guide is “work in progress” and an update will be published regularly.

You can help improving the content. We are looking forward to any type of feedback: questions, corrections, technical insights, problem cases, user experiences, or requests for additional topics to be covered.

Please, send your comments directly to bernd.kohler@sap.com!

INTRODUCTION

What's new about ABAP Core Data Services

ABAP Core Data Services¹ (CDS) is a data dictionary infrastructure that was introduced with SAP NetWeaver 7.40. It allows for defining and consuming semantically rich data models. The underlying principles are depicted in Figure 1.

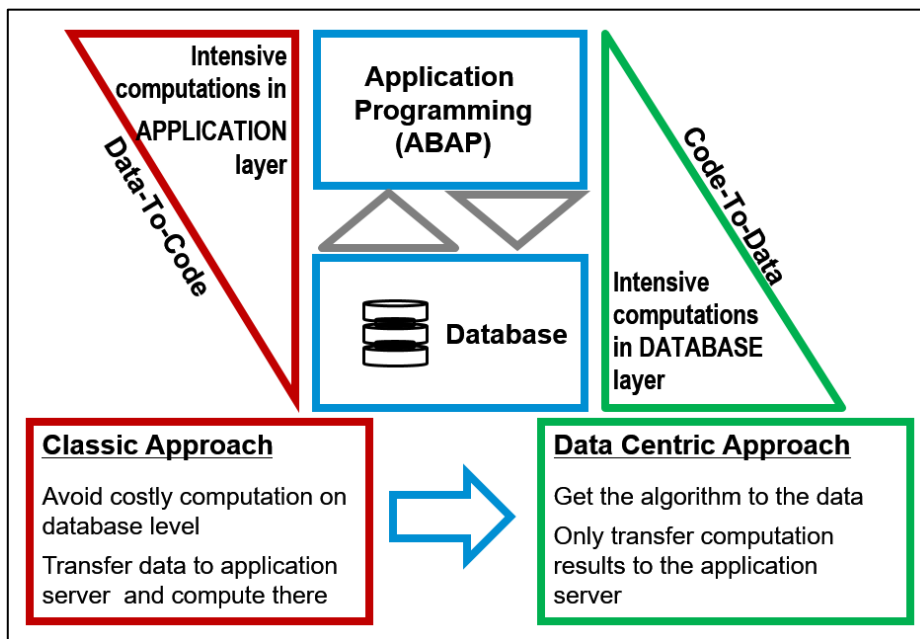


Figure 1: Principles of ABAP Core Data Services

Without CDS (labelled as “Classic Approach” in Figure 1), intensive calculations are done on the application layer avoiding costly computations in the database. This results in rather simple SQL queries between application and database layer. The drawback is however that lots of data needs to be transferred back and forth between those two layers. Often, this is very time-consuming.

CDS allows for a data centric approach. Intensive computations are pushed into the database by using complex views and functions (see Section “CDS features vs. SAP Release” on page 8). Depending on the use case, this may dramatically reduce execution time and simplify application coding as compared to the traditional approach.

This is especially improving performance for calculations that are close to the data, by reducing the data volume of communication between the application server and the database.

As shown in Figure 2, the major purpose for pushing down logic to the database is to spare on communication costs between the application server and the database server.

¹ **ABAP Core Data Services** should not be mixed up with **SAP HANA Core Data Services**. The latter serve to build design-time data-persistence models in SAP HANA Extended Application Services (SAP HANA XS) – for both XS classic and XS advanced models (see <https://help.sap.com/viewer/search?q=hana%20cds>).

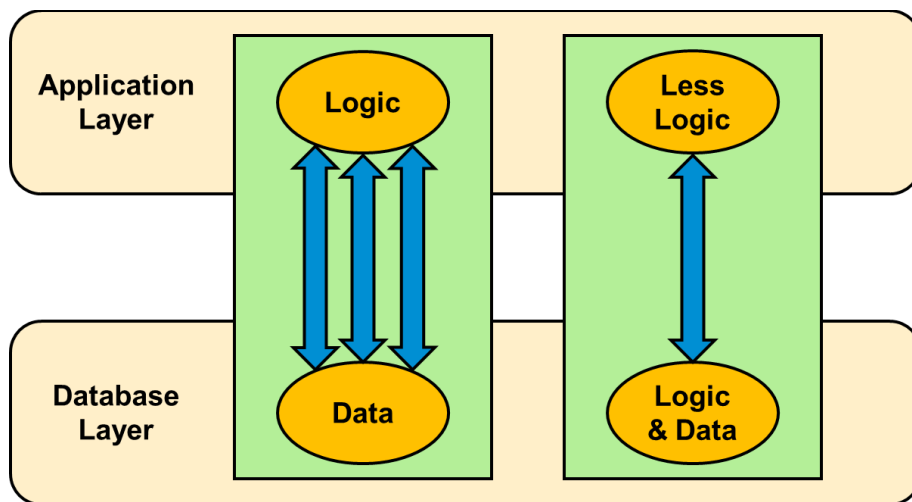


Figure 2: Reduction of communication costs

About this Guide

Purpose

The CDS framework was initially introduced to leverage the computational power of HANA DB. Nevertheless, it can also be used with all other databases that support SAP NetWeaver. This guide gives hands-on information on how to implement, run and optimize CDS based applications on various database products.

A great deal of online documentation covering CDS is already available. Whenever possible, the guide will reference these sources rather than replicate the related information.

Note that some links are only accessible with an SAP User Account (also known as S-user ID). SAP administrators in your company are usually able to provide one (without additional privileges).

Audience


This document is useful for the following audience:

- **IT architects** – to design CDS based applications (data structure, database layout, queries, etc.) and guideline application development
- **Application developers** – to understand database capabilities and potential limitations related to CDS
- **Software quality engineers** – to understand typical issues with ABAP CDS and tools for mitigating them
- **SAP Basis administrators** – to ensure the required SAP maintenance level
- **Database administrators** – to provide the correct database maintenance level and optimize the database performance

Scope

The following database products supported for SAP Business Suite are covered:

- IBM Db2 for i
- IBM Db2 for Linux, Unix and Windows
- IBM Db2 for z/OS
- Microsoft SQL Server
- Oracle Database
- SAP Adaptive Server Enterprise (SAP ASE)

- 
- SAP HANA DB
 - SAP MaxDB

Many technical details and recommendations apply to all database types. Individual information is covered in Section “Database Specifics” (page 17ff.).

Structure and Content

The guide is made up of three parts:

- 1) Section “ABAP Core Data Services” provides a technical overview and describes the CDS framework. In addition, it lists some of the services provided on top of CDS and shows how it is used in SAP applications.
- 2) Section “Best Practices” gives general recommendations on how to deal with CDS within SAP solutions and home-grown applications.
- 3) Section “Database Specifics” lists database specific recommendations and requirements when using CDS. Maintenance levels, settings and tuning measures are clarified.

ABAP CORE DATA SERVICES

Availability and Core Capabilities

ABAP CDS became available first with SAP NetWeaver 7.4 SP05. Its benefits can be summarized as follows:

- 1) **Semantically rich data-models** – is declarative and close to conceptual thinking (refer to <https://tinyurl.com/SAP-ABAP-CDS-Data-Model> for details)
- 2) **Completely based on SQL** – offers many standard SQL features like joins or build-in functions
- 3) **Compatible across databases** – is available with all database products that support SAP NetWeaver 7.40 and higher
- 4) **Annotations** - supports domain-specific annotations which can be easily evaluated by other components, such as UIs, analytics, or OData services
- 5) **Associations** – enables simple definition of views on top of views and path expressions to navigate along relations
- 6) **Extensibility** – allows simple extension of CDS views with fields either on model level through extensions or on meta-model level through annotations

A detailed introduction to ABAP CDS is available at <https://tinyurl.com/SAP-ABAP-CDS-Introduction>.

CDS features vs. SAP Release

Many CDS specific enhancements have been added to SAP NetWeaver after the framework was initially released with SAP NetWeaver 7.40 SP05, among them:

- SQL joins: INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, CROSS JOIN
- SQL SET operations: UNION, UNION ALL
- SELECT clauses: WHERE, GROUP BY, HAVING, AS
- Literals, arithmetic operators, conditional expressions: +, -, *, /, NOT, AND, OR, BETWEEN, =, ...
- Aggregate functions: AVG, MAX, MIN, SUM, COUNT
- Numeric functions: CEIL, MOD, ABS, DIV, DIVISION, FLOOR, ROUND, FLTP_TO_DEC
- String functions: SUBSTRING, LPAD, CONCAT, CONCAT_WITH_SPACE, REPLACE, INSTR, LEFT, LENGTH, LTRIM, RIGHT, RPAD, RTRIM, UPPER, LOWER
- Byte string functions: BINTOHEX, HEXTOBIN
- Date and time functions: DATS_DAYS_BETWEEN, DATS_ADD_DAYS, DATS_IS_VALID, ...
- Special functions: CAST, COALESCE, CURRENCY_CONVERSION, UNIT_CONVERSION, DECIMAL_SHIFT, ...
- Session variables: `$session.user`, `$session.client`, `$session.system_date`, ...

The following blog provides an excellent overview about the availability of certain features:

<https://tinyurl.com/SAP-ABAP-CDS-Feature-Matrix>

SAP Help Portal also serves as a comprehensive source of information (see table below):

Documentation Links

SAP NetWeaver Release	SAP Applications	Link to SAP Help Portal	What's new
7.40	ERP 6 EHP 7	tinyurl.com/SAP-ABAP-CDS-Docu-740	
7.50	ERP 6 EHP 8 Banking Services 9 Solution Manager 7.2	tinyurl.com/SAP-ABAP-CDS-Docu-750	tinyurl.com/SAP-ABAP-CDS-New-in-750
7.51		tinyurl.com/SAP-ABAP-CDS-Docu-751	tinyurl.com/SAP-ABAP-CDS-New-in-751
7.52		tinyurl.com/SAP-ABAP-CDS-Docu-752	tinyurl.com/SAP-ABAP-CDS-New-in-752

Development Framework

CDS views are developed and maintained in ABAP in Eclipse (see Figure 3) whose textual editor provides a rich feature set for fast development including:

- Code completion
- Data preview
- Quick-fix function
- Syntax highlighting
- Dependency analyzer
- Display of CREATE statement

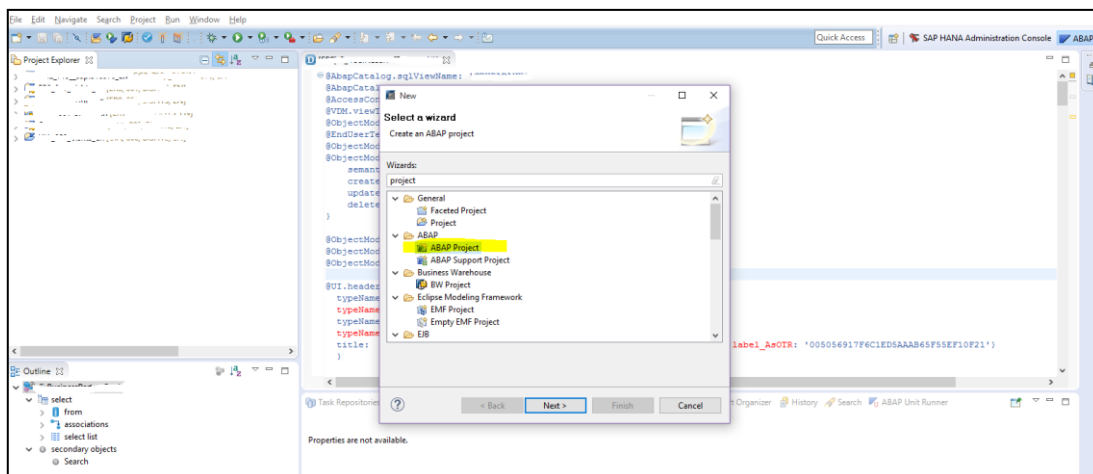


Figure 3: ABAP in Eclipse development framework

There is lots of information available on how to develop CDS based applications by using the ABAP in Eclipse framework. Refer to the following links:

- **Configuration:** http://help.sap.com/download/netweaver/adt/SAP_ADT_Configuration_Guide_Backend_en.pdf
- **Guides:**
 - <https://tinyurl.com/SAP-NW750SP16-CDS-User-Guide>
 - <https://tinyurl.com/SAP-NW750SP16-ABAP-in-Eclipse>
- **Tutorials:**
 - <https://developers.sap.com/group.abap-cds.html>
 - <https://www.sap.com/developer/tutorials/abap-create-project.html>

- <https://www.sap.com/developer/tutorials/abap-dev-adt-create-cds-view.html>
- <https://blogs.sap.com/2016/09/26/core-data-services-cds-in-sap-s4-hana>

Services Based on CDS Views

The full potential of the CDS framework is only realized by using the services built on top of it. The following sections serve as an introduction to some of them.

OData Clients

CDS views can be easily exposed via OData by adding annotation `@Odata.publish: true` to the view definition. Figure 4 illustrates the components that participate in the exposure process. The view activation process generates several service artefacts. Subsequently, these services are added to the SAP Gateway service catalog (using transaction `/IWFND/MAINT_SERVICE`) and become thus available for consumption by an OData client (e.g. an SAP Fiori app).

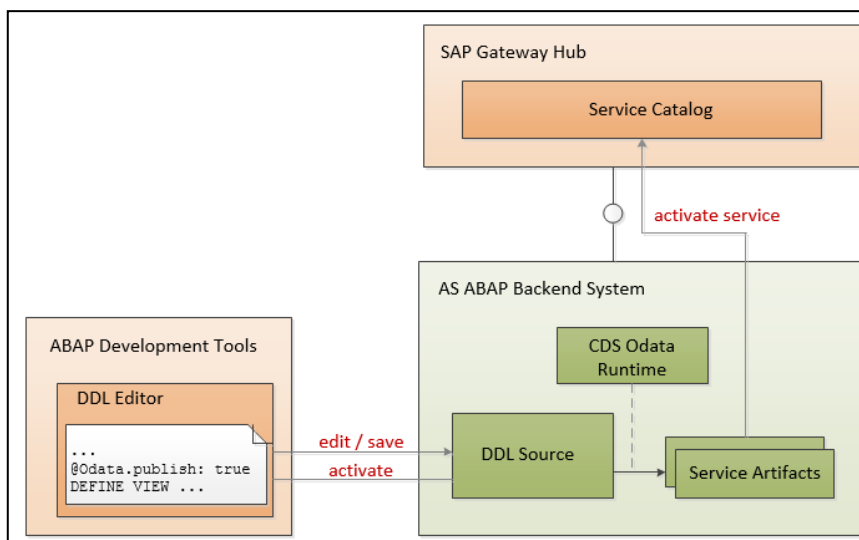


Figure 4: Components and activities when exposing CDS views to OData

More information is available here:

- <https://tinyurl.com/SAP-ABAP-CDS-and-OData>
- <https://blogs.sap.com/2015/04/20/creating-odata-services-out-of-cds-views>
- <https://blogs.sap.com/2018/08/13/beauty-of-odata>

SAP Fiori

Based on the OData exposure of CDS described above, it is then rather straightforward to create an SAP Fiori app using the development framework SAP WEB IDE (either locally or within SAP Cloud Platform). As depicted in Figure 5 the SAP Fiori User Interface connects to SAP Gateway using the OData services.

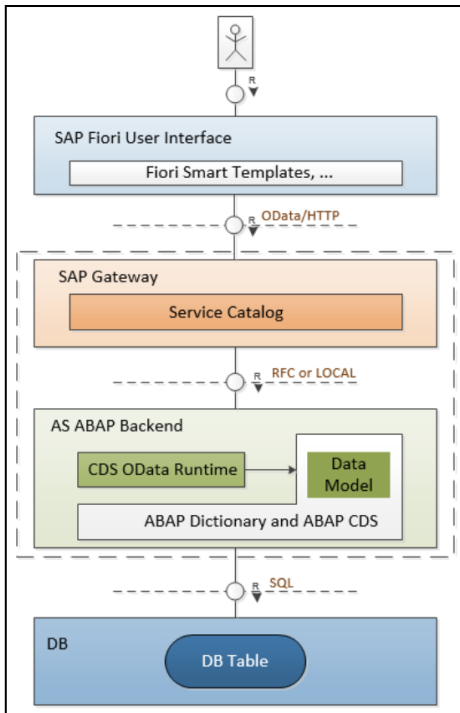


Figure 5: CDS consumption by SAP Fiori - architecture overview

A detailed step-by-step description on how to create such an app is provided here:

1. <https://tinyurl.com/SAP-ABAP-CDS-Create-Fiori-App1>
2. <https://tinyurl.com/SAP-ABAP-CDS-Create-Fiori-App2>

Lots of information is available at the SAP Community Wiki for SAP Fiori:

- <https://wiki.scn.sap.com/wiki/display/Fiori/All+Things+SAP+Fiori>

SAP BusinessObjects Analysis for Microsoft Office

SAP BO Analysis for MS Office is a business intelligence (BI) and multi-dimensional data analysis software. It allows to filter and manipulate data, to identify trends and outliers right within Excel, and to share findings in live PowerPoint presentations (Figure 6 shows an example).

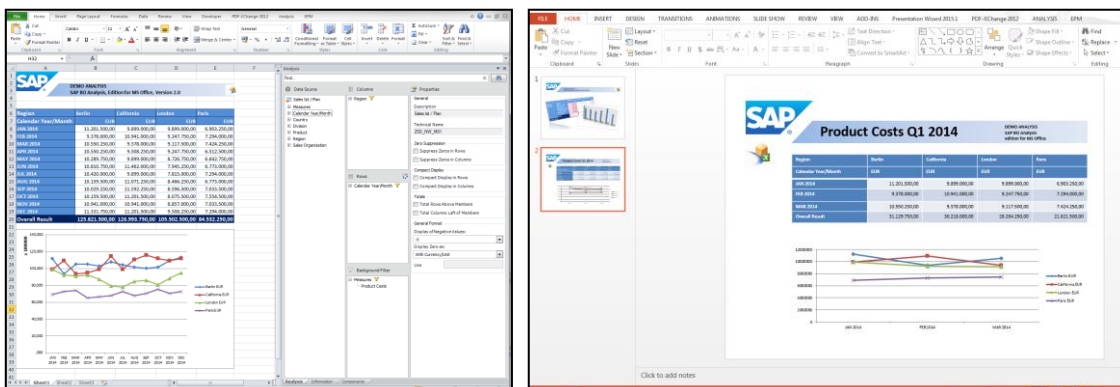


Figure 6: Analysis for Microsoft Office with Excel and PowerPoint

CDS views can easily be added as a data source, for more information refer to:

- <https://blogs.sap.com/2017/05/12/consumption-of-cds-views-in-analysis-for-office>

- <https://tinyurl.com/SCN-Analysis-for-MS-Office> (Product tutorials)

BW Context

ABAP CDS views can also be used within an SAP BW context.

Refer to the following link for details: <https://tinyurl.com/SAP-ABAP-CDS-and-BW>

SAP Applications using CDS

There is a variety of SAP applications based on CDS. The following sections gives a brief overview (without claim of completeness).

ERP 6.0 EHP8

The latest version of SAP ERP, which is SAP ERP 6.0 Enhancement Pack 8, is delivered with more than 2300 CDS views. One of the ERP components that benefit from using CDS views is for instance SAP Commodity Management (see <https://tinyurl.com/SAP-Commodity-Mgmt-608SP13>).

Rapid Replenishment Planning

Replenishment is a method of supplying recipients (sites or external customers) with merchandise on a demand-driven basis. In replenishment planning, requirements are calculated using the current stock situation. Once this has been done, follow-on documents (for example, purchase orders or sales orders) are generated for the supply of merchandise.

In rapid replenishment planning, the system uses an optimized method for determining the required elements (stock, forecasts, receipts/issues) and calculating the required quantity. It enables – depending on the database system used, and the concrete attributes of the planning run – a significant runtime saving in comparison with classic planning.

For details refer to [SAP Note 2051280](#) and <https://tinyurl.com/SAP-Retail-Rapid-Replmnt-60811>.

Convergent Invoicing

SAP Convergent Invoicing is a finance application that integrates the following applications:

- SAP Convergent Charging (see https://help.sap.com/viewer/product/Convergent_Charging)
- SAP CRM (see <https://tinyurl.com/SAP-Customer-Relationship-Mgmt>)
- SAP Contract Accounts Receivable and Payable
(see https://help.sap.com/viewer/product/SAP_CONTRACT_ACCOUNTS_RECEIVABLE_AND_PAYABLE_FOR_SAP_ERP)

The comprehensive integration enables business processes from the consuming a service through pricing and billing right up to dispatching of the invoice to the customer.

On the front-end site, SAP Convergent Invoicing exploits SAP Fiori. On the backend, complex queries are executed.

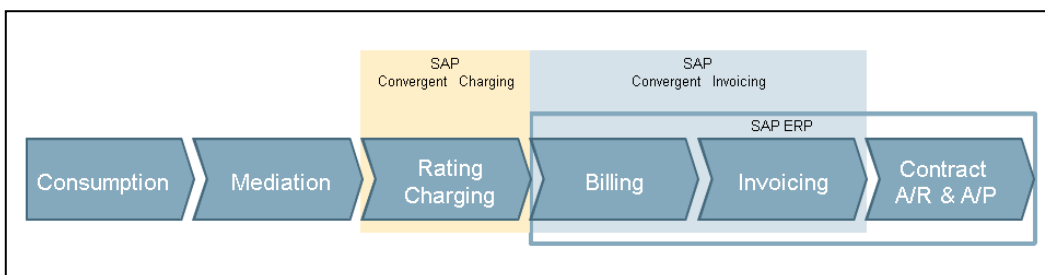



Figure 7: SAP Convergent Invoicing



More information is available at: <https://tinyurl.com/SAP-Convergent-Invoicing-61813>

Banking Services from SAP

SAP banking services from SAP 9.0 provides a variety of new Fiori-like apps. Two of those apps heavily rely on CDS implementations:

- Get Bank Customer Overview (see <https://tinyurl.com/GetBankCustomerOverview-9SP08>)
- Manage Time Deposits (see <https://tinyurl.com/ManageTimeDeposits-9SP08>)

BEST PRACTICES

SAP Applications

SAP applications using CDS views are extensively checked and tested on your database by SAP before being released and shipped. So, the related transactions run out-of-the-box.

In addition, we recommend the following:

- 1) **Check the database:** Lift the underlying databases to the minimum versions and patch levels specified in Section “Database Specifics” (page 17ff.). Also, ensure that the database settings are according to the recommendations mentioned there.
- 2) **Check SAP Release Notes:** There is database related information available for some SAP applications using CDS views. See Section “SAP Applications” (page 17) for details.
- 3) **Update the SAP Patch Level:** Apply all SAP Patches and Notes recommended for your database. Refer to Section “Database Specifics” (page 17ff.) for details.
- 4) **Keep an eye on CDS based workload:** Regularly monitor the performance and workload impact of CDS based applications, e.g. by using the ABAP SQL Monitor (see <https://tinyurl.com/ABAP-SQL-Monitor> and <https://tinyurl.com/SQL-Performance-Monitor-75SP16>).
- 5) **Establish a remote support connection:** In case SAP support is needed for the analysis of CDS related problems, the required support infrastructure should already be in place. Refer to Section “Incident Support” (page 17) to set it up.

In case of performance issues, refer to the database specific “Tuning Measures” within Section “Database Specifics” (page 17ff.). If that fails open an incident using component **HAN-DB** (for HANA DB) or **BC-DB-<DBS>** (<DBS>=DB2, DB4, DB6, MSS, ORA, SDB, SYB).

Custom Applications

The benefits of the CDS framework laid out in Section “ABAP Core Data Services” (page 8) are by no means only restricted to SAP applications. For customers, home-grown applications are an essential part of their SAP related portfolio and workload. Many of these apps could significantly benefit from using CDS features.

For custom development, the following recommendations apply:

- 1) **Start small:** It is important to gain in-house experience with the “new” database centric development model as well as the performance and life-cycle management of CDS views. Therefore, start with a small project that does not affect business critical transactions and gradually expand the footprint of CDS within your SAP landscape.
- 2) **Check the database:** Lift the underlying databases to the minimum versions and patch levels specified in Section “Database Specifics” (page 17ff.). Also, ensure that the database settings are according to the recommendations mentioned there.
- 3) **Update to SAP NetWeaver 7.50:** Although CDS has been made available already with SAP NetWeaver 7.40, it is advisable to have your SAP landscape updated to SAP NetWeaver 7.50 (which corresponds to Business Suite 6 Enhancement Pack 8) before starting CDS based developments. Also, apply all SAP Patches & Notes recommended for your database; refer to Section “Database Specifics” (page 17ff.) for details.
- 4) **Implement a governance process:** To avoid any surprises once CDS views have arrived in the productive environment, you should establish clear rules and checks for the development process. More details are covered in Section “Development Rules and Governance” (page 15).
- 5) **Ensure quality assurance:** Testing with realistic data on all database types present in your SAP landscape is a must for newly developed CDS-based applications. It is highly recommended to specify annotations with respect to service qualities, data sizes, data categories and VDM view type so that related static checks could be used.
- 6) **Keep an eye on CDS based workload:** Regularly monitor the performance and workload impact of CDS based applications, e.g. by using the ABAP SQL Monitor (see <https://tinyurl.com/ABAP-SQL-Monitor> and <https://tinyurl.com/SQL-Performance-Monitor-75SP16>).

- 7) Establish a remote support connection:** In case SAP support is needed for the analysis of CDS related problems, the required support infrastructure should already be in place. Refer to Section “Incident Support” (page 17) to set it up.

In case of performance issues, refer to the database specific “Tuning Measures” within Section “Database Specifics” (page 17ff.).

Development Rules and Governance

This section lists recommendation on how to pursue in-house application development (also known as “custom development”) using CDS capabilities.

The CDS development framework offers enormous possibilities. However, the laws of physics and computer science still apply.

Therefore, we advise to stick to some basic development rules:

1. General Rule

Use SQL best practices when using CDS views.

2. OLTP and OLAP

Separate data models for analytic and transactional purposes.

3. DB Limits

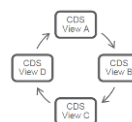
Pay attention to database limits (Section “Database Specifics” 17ff.).

4. Complexity

- Build entity-relationship models before defining CDS views. Take a careful decision on normalized models vs. redundant storage of data, in particular with respect to avoiding **JOIN** operations on huge database tables.
- Restrict the complexity of CDS views depending on their usage:

View category	Usage	Number of tables	Functions	Aggregation	Cyclic associations or joins
1	Within business logic of high-volume transactions	≤ 3	No	No	No
2	Within business logic of transactions	≤ 5	Only for result set	No	No
3	Within UI for single object retrieval	≤ 10	Yes	Yes	No
4	Analytical reporting	Open	Yes	Yes	Yes

- Avoid cyclic references in the definition of CDS views (to prevent problems during activation).



5. Performance

- Create indexes on base tables to enable index-only access (not relevant for HANA DB).
- HANA only: For secondary indices on HANA refer to SAP Note [1794297](#). Additional indexes can be helpful in tables with many entries (> 1 million) for selective fields that are used as filter of frequently executed queries. HANA secondary indices are predominantly single column indices.
- SFIN on HANA only: Avoid access to compatibility views for performance-intensive applications (if runtime is critical or if frequently executed).

- Avoid calculated fields in **WHERE**-clauses and **ON**-clauses of CDS views, especially when tables with many entries are involved. Calculated fields result from functions like string functions (**CONCAT**, **RTRIM**, ...) and **CASE** expressions. Note that all fields provided by a table function act as calculated. For a calculated field to be used (as filter or join condition) the values of this field for every row must be evaluated at this point of processing. That may be very time-consuming.
- Avoid constants in **ELSE** branches within **CASE** especially for views to be reused in joins as it potentially limits DB optimization due to not preserving the null value. The same limitation applies to fields defined as constant if the view in which they are defined is on the right side of a **LEFT OUTER JOIN**.
- Avoid cyclic associations or cyclic joins with involvement of left outer join, especially when tables with many entries are involved.
- To achieve a certain business purpose, choose the simplest CDS view possible. Do not create the all-in-one complex view covering all demands.
- Use associations to improve performance: If they are just defined and exposed within the projection list without direct exposition of fields, they are pure metadata; only if used in a path expression they become standard joins.
- If CDS views contain **LEFT OUTER JOIN** relationships between tables, use the **MANY TO ONE** clause if possible. Check if **UNION** clauses could be safely replaced by **UNION ALL** without changing the semantics.
- For HANA DB, prefer **UNION [ALL]** over **CASE** expressions in the models as they allow more room for preferable choices for the database optimizer.

6. Life-cycle management

- Ensure an identical SAP maintenance level on all involved SAP systems (development, Q&A, production).
- Do not rename generated views after their release to the SAP landscape.

7. Approval Process

Review your approval processes before starting CDS based development.

Consider the following:

- Communicate the development rules lined out in this section.
- Review and approve complex CDS hierarchies before releasing them to the SAP landscape.
- Ensure that new CDS workloads do not impair important business processes in the same system (review of workload management, capacity planning, etc.).

Performance Safeguarding

In case of performance issues related to the database refer to the database specific “Tuning Measures” within Section “Database Specifics” (page 17ff.).

The following blogs introduce into the performance of CDS views especially on HANA DB and described how to safeguard it:

- <https://blogs.sap.com/2018/03/02/performance-of-abap-cds-on-sap-hana-at-sap-teched-2017>
- <https://tinyurl.com/1-CDS-View-Complexity>
- <https://tinyurl.com/2-SQL-Optimizer-Plan-Cache>
- <https://tinyurl.com/3-Rules-for-Good-Performance>

Many recommendations given in these blogs are also valid for the row-based database types supported by SAP, i.e. IBM Db2, Oracle, MS-SQL, SAP ASE & SAP MaxDB.

DATABASE SPECIFICS

General Remarks

Hardware Requirements

Standard sizing with SAP Quick Sizer applies. CDS views may pose rather individual memory and CPU requirements to the infrastructure depending on their complexity and the data volume. Tuning can only be done hands-on. Refer to the DB specific Sections “Tuning Measures” for hints on how to improve the performance of CDS based queries.

SAP Applications

Database specific information is available for CDS enhanced SAP applications. Please, refer to the notes listed below:

SAP Note	Title / Comment
<u>2051280</u>	Rapid replenishment planning
<u>2429167</u>	Usage of account search in Fiori app for banking services from SAP - performance on different data bases
<u>2429180</u>	Usage of business partner search in Fiori apps for banking services from SAP - performance on different data bases
<u>2311347</u>	Database-Related Performance Information for Snapshot-Based Mark-to-Market Reporting
<u>2136479</u>	Database-Specific Information for SAP Convergent Invoicing

General Notes

SAP Note	Title / Comment
<u>2239157</u>	Support of database features

Life-Cycle Management

The following recommendations apply if you plan to migrate an SAP system with self-defined CDS views to a different database:


- Check the notes listed above in Sections “SAP Applications” and “General Notes” for remarks relevant for the target database.
- Verify that all self-defined CDS views in the source systems comply to the limitations of the target database (see the databases specific Sections “DB Limits” below).
- Perform a proof-of-concept migration of the productive system and check the performance of all custom applications based on CDS views.

Incident Support

In case of persistent database issues with a CDS Query, create an incident in the appropriate SAP database component **HAN-DB** (for HANA DB) or **BC-DB-<DBS>** (<DBS>=DB2, DB4, DB6, MSS, ORA, SDB, SYB).

For analyzing CDS related issues, an SAP R/3 standard support connection is not enough as access by ABAP Development Tools outside the SAPGUI are required. To ensure that SAP support has full access to the system for problem determination, enable one of following remote access options:

- WTS Connection to a desktop with the ABAP development tools installed

- 
- R/3 Support and SAP NI Connection to allow remote connection of ABAP Development Tools by SAP Service or Development staff

Details on SAP service connections are available at <https://tinyurl.com/SAP-Remote-Service-Connections>.

Also, ensure that support users are authorized to execute the following transactions:

- DBACOCKPIT, ST04
- SQLMD
- ST03N
- SAT, ST05, ST12

For HANA DB, the authorization for transaction DBACOCKPIT in combination with ST05 and ST12 is particularly helpful because it allows PlanViz files to be saved – in case Plan Visualizer (PlanViz) is enabled in the system.

IBM Db2 for i

DB Levels

We recommend that you use IBM i 7.2 or higher to get the best possible performance. In general, applying the product temporary fixes (PTFs) listed in the IBM Info APAR for SAP for your database release will ensure that you have all PTFs that are recommended to run SAP. You can use the health check functionality of report **RSDB4CDST00LS** to check for individual PTFs beneficial to run CDS. Those include fixes to known bugs as well as performance features like **LEFT OUTER TO ONE JOIN**, table function inlining, **LIMIT – OFFSET**, and others. All PTFs are also listed in [SAP Note 2075068](#) ("IBM i: Overview of CDS on DB2 for IBM i").

DB Version	SAP Note	Title / Comment
IBM i 7.2	2075068	SAP Support Required PTF List for IBM i 7.2
IBM i 7.3	2075068	SAP Support Required PTF List for IBM i 7.3
IBM i 7.4	2075068	SAP Support Required PTF List for IBM i 7.4

DB Settings

IBM Db2 for i is offering a variety of tools and advanced tuning measures that can help to speed up the execution of specific CDS queries. Among those are:

- **IBM i Access Client Solutions (ACS)** is a Java based platform-independent tool that you can download to your workstation from this link: <https://www.ibm.com/systems/power/software/i/access/solutions.html>. With ACS, you can connect to any IBM i server in your landscape and perform administrative tasks. For IBM Db2 for i, the tool offers options to work with database objects in schemas, run SQL scripts and analyze the database performance. You can also configure the plan cache size with this tool. IBM i Access Client Solutions and the SAP transaction **DBACOCKPIT** offer similar, in some areas partially overlapping functions.
- **Query Options File QAQQINI:** A variety of database options can be configured through a query options file named QAQQINI. Depending on the location of the file and the configuration of your SAP systems, the configured parameters in this file can take effect system-wide, per SAP system or for selected SQL statements within an SAP system. See [SAP Note 820325](#) for more information about the general usage of the query options file. Specific options to configure for certain purposes are documented in separate SAP Notes that deal with selected use cases.
- **Db2 Symmetric Multiprocessing (SMP)** allows the SQL optimizer to split up a single query into multiple threads that are executed in parallel. This may speed up the processing of a query on a large table, but at the same time uses more processing resources, so it should only be used when enough resources are available. To use Db2 Symmetric Multiprocessing, option 26 of the operating system must be installed. [SAP Note 1879983](#) explains how to configure the use of SMP in an SAP landscape, for example through a conditional statement hint for a specific CDS view.
- **Encoded Vector Indexes (EVIs)** offer an alternative to the common binary radix tree indexes and are a variation on bitmap indexing. They provide benefits to JOIN operations, specifically in star or snowflake schemas as they are used in business warehouse queries. They may also show significant improvements when only a few columns of a very large table are queried and single-column EVIs exist over all requested columns. In this case, the SQL optimizer can retrieve all the required data for the output from the EVIs without the need to access the table itself. This access method is called EVI Only Access (EOA), and it is available by default as of IBM i 7.2. More information about using EVIs with SAP on IBM Db2 for i can be found in [SAP Note 2588130](#).
- **Preloading Database Objects into a Main Storage Pool:** In general, the storage management functionality of the IBM i operating system and Licensed Internal Code are automatically taking care of assigning main storage to SAP systems and database operations. By default, all SAP systems within a logical partition of a server share a common main storage pool. However, there are cases when you want to separate workloads and assign specific main storage pools to certain SAP systems (see [SAP Note 49201](#)).

It may improve the performance of CDS views which access large tables, if you can use EVI Only Access (EOA) and preload the EVIs into a separate main storage pool prior to the first execution of the statement. The necessary configuration steps are explained in [SAP Note 2588130](#).

The following table concludes all SAP Notes mentioned above:

DB Version	SAP Note	Title / Comment
Any	820325	IBM i: Use of QAQQINI with SAP
≥ IBM i 7.2	1879983	IBM i: Support for symmetric multiprocessing (SMP)
Any	2588130	IBM i: Using Encoded Vector Indexes with SAP
Any	49201	IBM i: Main memory settings for SAP

DB Limits

After view expansion into a single statement, the number all table references must not be larger than 225 (see [SAP Note 2422860](#)). If the same table is referenced more than once in a view, each reference of the table counts towards that limit. The SAP-enforced limit of 225 table references in a view is smaller than the database limit of 255 on IBM Db2 for i.

SAP Notes & Patches for CDS

[SAP Note 2075068](#) provides a list of all SAP notes relevant to successfully run CDS on IBM i. The list comprises kernel and ABAP patches to fix known bugs as well as to activate new database performance features. For convenience, you can run the health check functionality of report **RSDB4CDSTOOLS** telling you what exactly is missing in a system.

For advanced tuning, it might be beneficial to install and activate the new DBA Cockpit.

SAP Note	SAP BASIS	Title / topic
2075068	≥ 740	IBM i: Overview of CDS on DB2 for IBM i
2033326	all	IBM i: Enhancements in the DBA Cockpit for the SAP Database Performance Collector
2142186	≥ 740	IBM i: Performance: Fast replenishment planning/rapid replenishment planning
2422860	≥ 740	CDS View Complexity

SAP Life-Cycle Management

When renaming the database, the schema information in certain CDS object types may not be updated correctly, thus causing SQL errors at runtime when using these CDS views. In such a case, CDS views need to be created again through selected ABAP programs. See [SAP Note 2368628](#) for a description of the ABAP programs to execute and the cases where manual recreation is needed.

If you created EVIs to help the performance of CDS views, the EVIs will not be transported within your system landscape. To achieve the same, good performance on all your systems, you must create the EVIs manually in your test or production system after transporting the other changes.

Tuning Measures

Transaction	Parameter / Metric to be checked	Activity / Comment
IBM i ACS → SQL Performance Center → Plan Usage Summary → Current Plan Cache Size Threshold	Maximum Plan Cache Size	Compare to "Current Plan Cache Size". If possible, select "Change Configuration" and set to "*AUTO", which will keep the hit ratio at 90%.
DBA Cockpit → Performance → SQL Statements	Expensive SQL statements	Select most expensive SQL statements for analysis. Use function Explain to look at access plan and advised indexes for a particular statement.
DBA Cockpit → Diagnostics → EXPLAIN (new version)	Access plan for a particular SQL statement	Enter SQL statement for analysis, then check and analyze expensive nodes.
DBA Cockpit → Performance → SQE Indexes Advised	Index advices that the SQL optimizer found to be useful. Advices cannot be mapped to specific SQL statements.	Check Number of rows (table size), runtime of most expensive query and timestamp of Last advised before creating an advised index. Test performance afterwards. Not all indexes advised are necessarily needed or will be used.
DBA Cockpit → Performance → Advised Indexes	Summary of all index advices for all SQL statements in the plan cache, grouped by table. Advices can be mapped to specific SQL statements.	Check table size, total run time and average result set size to identify useful indexes. Display all statements for the selected table and check if multiple index advisories can be combined into a single index, so that the number of indexes in the database does not grow too much.
DBA Cockpit → Diagnostics → DB Catalog Browser	View definition / Object dependencies / #table references	Understand and simplify CDS view definitions and object interdependencies of complex CDS views. Identify large tables as potential candidates for EVIs.

For more information on how to use the DBA Cockpit, search for "CCMS: IBM DB2 for i" at <http://help.sap.com>. You can limit your search to the product "SAP NetWeaver" at the requested release level. For example, with SAP NetWeaver 7.5, you will be directed to <https://tinyurl.com/SAP-NW750-CCMS-Db2-for-i>.

In case of problems or questions, you may always open an incident using component **BC-DB-DB4**.

IBM Db2 for Linux, Unix and Windows

DB Levels

DB Version	SAP Note	Title / Comment
Any	101809	Supported Db2 Versions and Fix Pack Levels
Db2 10.5 for LUW	1851853	Using Db2 10.5 with SAP applications
Db2 11.1 for LUW	2303763	Using Db2 11.1 with SAP applications
Db2 11.5 for LUW	2751085	Using Db2 11.5 with SAP Applications

It is highly recommended to run at least with Db2 11.1m4fp4 or Db2 11.5 and the latest Fix Pack to benefit from Db2 optimizations specific to CDS.

DB Settings

DB Version	SAP Note	Title / Comment
Db2 10.5 for LUW	1851832	Db2 10.5 Standard Parameter Settings
Db2 11.1 for LUW	2303771	Db2 11.1 Standard Parameter Settings
Db2 11.5 for LUW	2751102	Db2 11.5 Standard Parameter Settings
Any	2047006	Use of Db2 SMP Parallelism (INTRA_PARALLEL=YES)

Use the DBA Cockpit to verify the correctness of the database and database manager configuration (*DBA Cockpit* → *Configuration* → *Parameter Check*).

DB Limits

All Db2 LUW limits are listed at:

- <https://tinyurl.com/IBM-DB2-LUW-11-1-Limits>
- <https://tinyurl.com/IBM-DB2-LUW-11-5-Limits>

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / Comment
2586972	≥ 740	Db2 11.1 enhancements for CDS (updated in January 2019 to cover Db2 11.1m4fp4), all enhancements for CDS are also contained in Db2 11.5
1818503	≥ 740	SAP Optimizer Profiles
868888	≥ 740	Use of DB2 Optimization Guidelines
150037	≥ 740	Database hints in Open SQL
2568486	≥ 740	performance problems with SELECT ... LIMIT ? queries
2465138		Replace function HEXTORAW with VARCHAR_BIT_FORMAT
2511399		performance of view with hexabin('000000000000000000000000000000')
2316918		LEFT OUTER TO ONE join in CDS views

SAP Life-Cycle Management

SAP Note	LC Mgmt. Task or Tool	Title / Comment
1365982		Current "db6_update_db/db6_update_client" script
2465138		Replace function HEXTORAW with VARCHAR_BIT_FORMAT
2324337	DDIC	CDS error during activation of CDS views with parameters in a SUM ZDO scenario
2237893	SUM	ABAP Upgrade with ZDO: Wrong DB Schema for CDS Objects

Tuning Measures

Ensure you are running on the latest Fixpack of Db2 as the product is constantly enhanced with additional features to optimize the execution of CDS workloads. Standard Db2 LUW tuning approaches are valid for CDS workloads as well. Ensure that the memory areas (e.g. Bufferpools and Sortheap) for Db2 are equipped with enough memory and CPU resources are available to support the parallel execution of queries. Also, ensure up-to-date database statistics along with the SAP-recommended setting of the database manager and the database configuration.

You can use the SAP DBA Cockpit to validate the configuration and to identify system bottlenecks, i.e.:

- *DBA Cockpit* → *Performance* → *SQL Cache*
- *DBA Cockpit* → *Performance* → *Buffer Pool*
- *DBA Cockpit* → *Performance* → *Sorts*

The following link connects to the DBA documentation, including a table with basic recommendations related to the performance: <https://tinyurl.com/SAP-on-IBM-Db2-LUW-Performance>

The key factors for good CDS performance are the following

- **Parallelism**

Best performance for CDS queries is achieved when executing a query with parallelism enabled and the optimal execution plan selected by the Db2 LUW optimizer.

You can verify the parallel execution of a query by using the explain functionality of the DBA Cockpit:

DBA Cockpit → *Diagnostics* → *EXPLAIN*

Please read through the following SAP Community document as an introduction for Explain functionality within the DBA Cockpit: <https://wiki.scn.sap.com/wiki/display/ABAP/Explain+Plan>

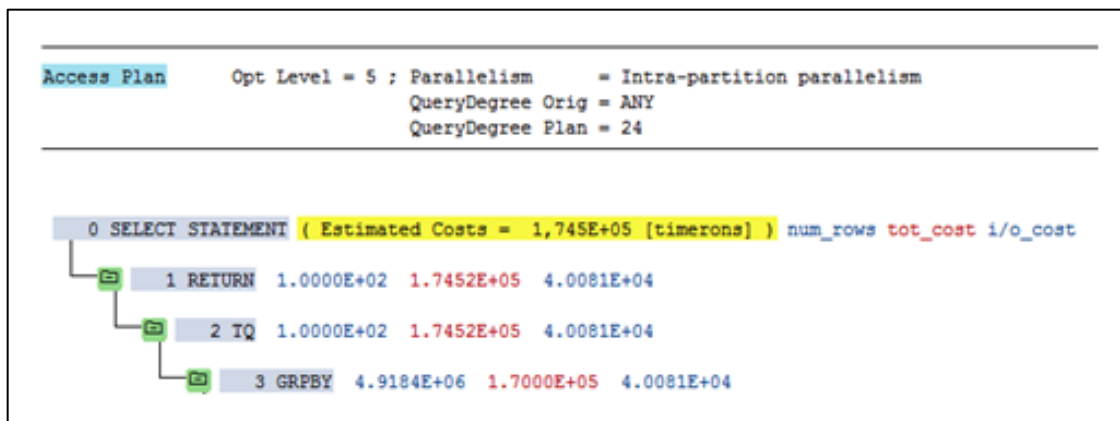


Figure 8: Db2 LUW Explain within the DBA Cockpit

Ensure that intra-partition parallelism is enabled with a meaningful number for QueryDegree. In addition, check the plan for the "TQ" (Table Queue) operator. This operator should be as high as possible in the plan as this is a good indicator for parallel execution as all operators above the TQ will be executed

without leveraging intra partition parallelism. Refer to Figure 8 which shows an Explain from the DBA Cockpit where the TQ operator sits at top of the access plan.

There are some operators that force the plan to be serialized like "Jump Scan" access to an index, so an additional index or a changed definition of a custom index can help. There are more operators that may force serialization. If you face a performance issue with a query and a low TQ operator, you may open an incident using component **BC-DB-DB6**.

- **Access Plan**

Beside the use of parallelism, the optimal access plan is the second important functionality to ensure good CDS Query Performance.

Check the access plan for expensive access to certain objects. Expensive means either a high number of timerons or many records processed within a certain leg of the plan. A good approach is to identify the critical path within the plan and identify the objects or operators that are causing a high number of timerons or are processing many rows. Often there is a table scan involved, and then additional indexes may help to improve query performance.

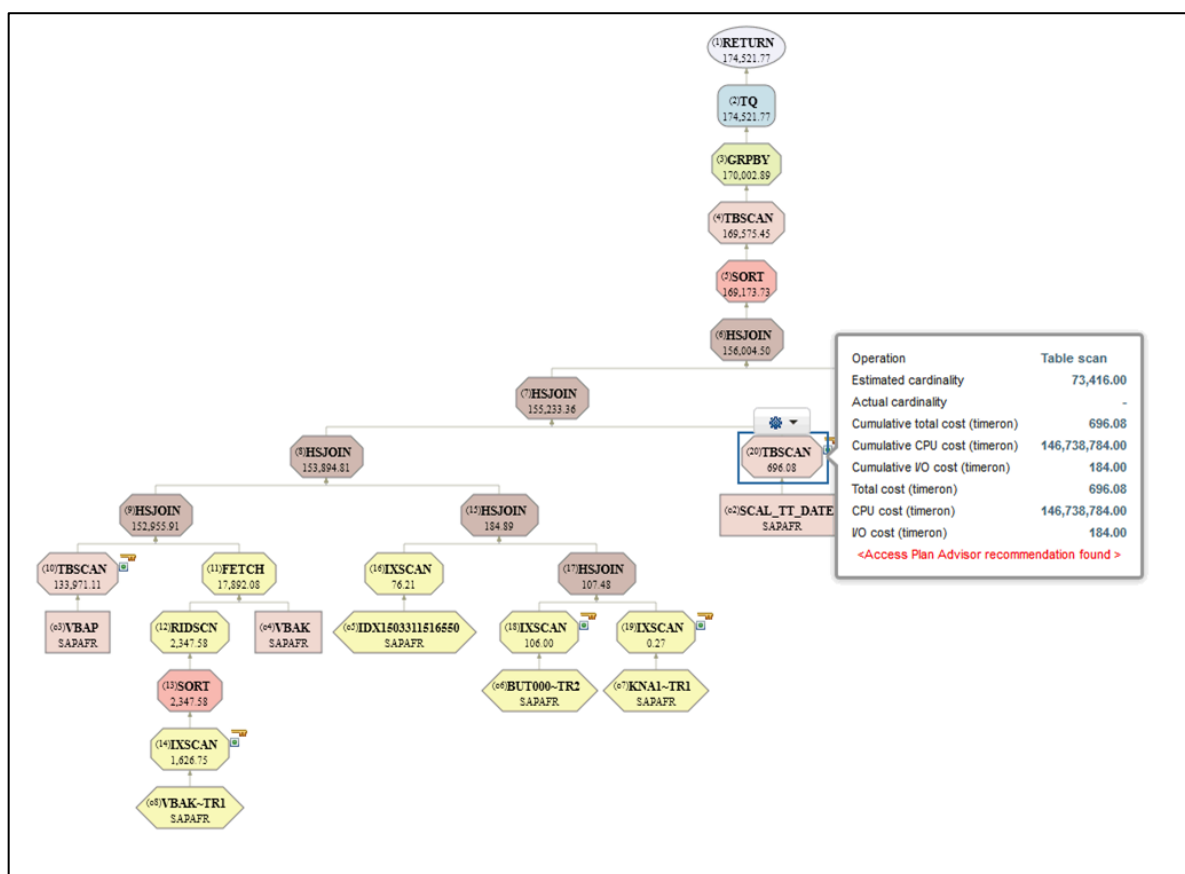


Figure 9: Db2 LUW Access Plan Graph from IBM Data Server Manager

Figure 9 shows the location of the TQ Operator as well as the information about a table scan together with a hint that a recommendation for this plan object is found. A similar functionality is available in the DBA Cockpit (*DBA Cockpit* → *Diagnose* → *Index Advisor*). Please, do not blindly apply the recommendations. Instead, verify the results first along with potential side effects – e.g. space consumption and insert overhead of additional indexes. Before applying such changes, you should also open an incident using component **BC-DB-DB6**.

- **Statistics**

The reason for non-optimal access plans could be missing statistics for the objects accessed in the plan. Please ensure that the Db2 runstats utility is running in automatic mode, which can be checked with *DBA Cockpit* → *Configuration* → *Database*, parameter **AUTO_RUNSTATS**. For a single table, you can also check

date and time of the last runstats with *DBA Cockpit* → *Space* → *Single Table Analysis*. From there, you can also refresh the statistics choosing "RUNSTATS in the background".

- **Cardinality**

As the CDS queries might be complex constructs with sometimes dozens of tables joined, the cardinality estimate of join results may be non-optimal and lead to wrong access plans. To verify such a mismatch in the estimated and actual data used for the access plan, you can use the Db2 utility "db2caem". This utility delivers a formatted EXPLAIN output, including section actuals (statistics for different operators in the access plan) and details like CPU time for the statement execution.

For details about this utility, refer to the following links within the IBM Knowledge Center:

- <https://tinyurl.com/IBM-DB2-LUW-11-1-db2caem>
- <https://tinyurl.com/IBM-DB2-LUW-11-5-db2caem>

- **Complexity**

As CDS queries can become complex, it is sometimes useful to identify the objects involved and their relationship or hierarchy. This may allow to isolate a problematic part of the query and eliminate many objects from the access plan, which helps to analyze the issue. You may use the SQL statement depicted in Figure 11 to retrieve an object hierarchy starting from a given root object.

DEPENDENT	OBJTYPE	FUNCTYPE
ZFS_CACCTHLDRSRC	V	-
ZFS_IACCTHLDRADD	V	-
ZFS_IACCTHLDR	V	-
BUT100	T	-
ZFS_IBUPADATA	V	-
ADRP	T	-
BP001	T	-
BUT000	T	-
TSTMP_CURRENT_UTCTIMESTAMP	F	S

Figure 10: Db2 LUW output of SQL listed in Figure 11

```

WITH
sysdependencies ( dschema, dname, dtype, bschema, bname, btype ) as
( select TABSCHEMA, TABNAME, DTYPE, BSCHEMA, BNAME, BTYPE
  from SYSCAT.TABDEP
  union all
  select ROUTINESCHEMA, SPECIFICNAME, 'F', BSCHEMA, BNAME, BTYPE
  from SYSCAT.ROUTINEDEP
  union all
  select VARSHEMA, VARNAME, 'v', BSCHEMA, BNAME, BTYPE
  from SYSCAT.VARIABLEDEP ),
depsearch ( dschema,
            dname,
            dtype,
            bschema,
            bname,
            btype,
            level,
            treepath ) AS
( SELECT distinct
  '' as dschema,
  '' as dname,
  '' as dtype,
  dschema as bschema,
  dname as bname,
  dtype as btype,
  0 AS level,
  cast( 'TOP' as varchar(4096)) AS treepath
FROM sysdependencies
WHERE dname = 'ZFS_CACCTHLDRSRC' -- edit object (e.g. view name) here
  and dschema = 'SAPMUK'        -- edit schema here
  and dtype = 'V'                -- edit object type here (v=view, F=routine)
UNION ALL
SELECT DEP.dschem,
       DEP.dname,
       DEP.dtype,
       DEP.bschem,
       DEP.bname,
       DEP.btype,
       CTE.level + 1 AS level,
       treepath || '->' || DEP.bname as treepath
FROM sysdependencies DEP, depsearch CTE
WHERE CTE.bname = DEP.dname AND
      CTE.bschem = DEP.dschem AND
      CTE.btype = DEP.dtype AND
      CTE.level < 100 )
SELECT
  substr(space(level*2)||bname,1,50) as dependent,
  btype as objtype,
  CASE
    WHEN btype = 'F' THEN (SELECT functiontype FROM syscat.routines
                           WHERE specificname = A.bname AND routineschema = A.bschem)
    ELSE NULL
  END as functype,
  CASE
    WHEN btype = 'F' THEN (SELECT CASE WHEN language = 'SQL' THEN valid ELSE NULL END
                           FROM syscat.routines
                           WHERE specificname = A.bname AND routineschema = A.bschem)
    WHEN btype = 'V' THEN (SELECT valid from syscat.views
                           WHERE viewname = A.bname AND viewschema = A.bschem)
    ELSE NULL
  END as valid,
  CASE
    WHEN btype = 'T' THEN (SELECT card FROM syscat.tables
                           WHERE tablename = A.bname AND tabschema = A.bschem)
    ELSE NULL
  END as card
FROM depsearch A
ORDER BY treepath

```

Figure 11: Db2 LUW SQL statement to retrieve an object hierarchy

Alternatively, you may again use IBM Data Server Manager, as this also shows the dependencies of objects in the database. It does not provide a complete hierarchy of objects, but you can easily navigate from object to object by drilling down either to depended objects or to parent objects that are referencing the current object.

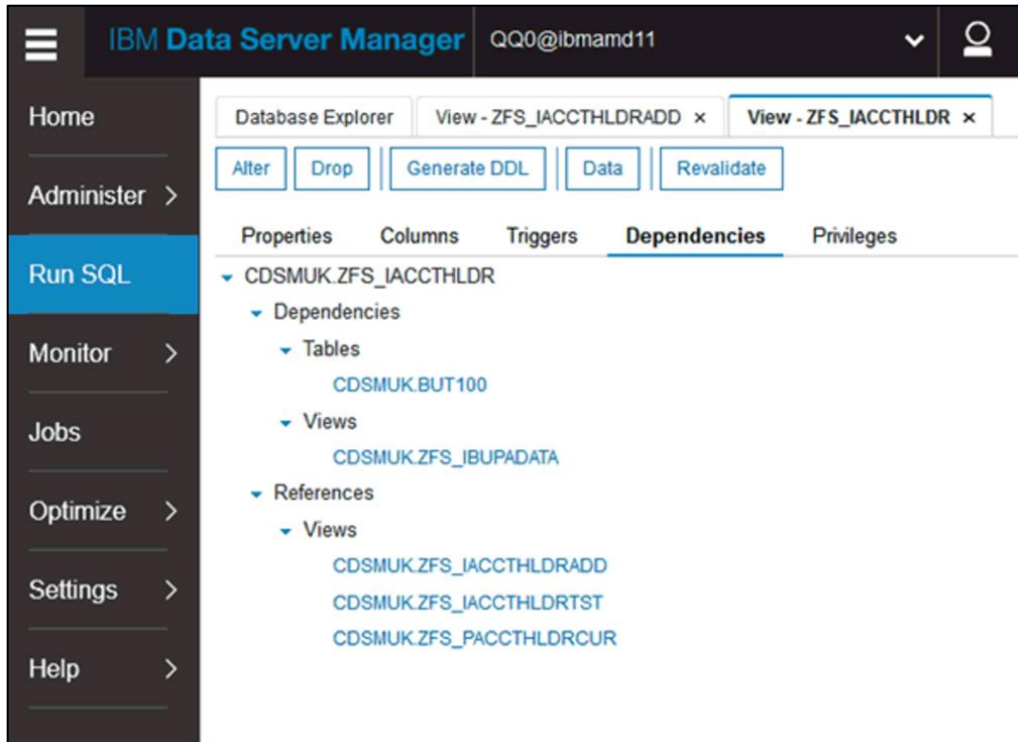


Figure 12: Db2 LUW dependencies and references for an object in the IBM Data Server Manager

IBM Db2 for z/OS

DB Levels

Version	SAP Note	Recommended Level
Db2 11 for z/OS	81737	PUT level 1804
Db2 12 for z/OS	81737	PUT level 1711 plus APARs PI80780 & PI89564
n/a	183311	Automated PTF Check

It is highly recommended to run with Db2 12 to leverage performance enhancements and avoid limitations (see “DB Limits” below). Use report **RSDB2FIX** to check that all recommended product temporary fixes (PTFs) are applied ([SAP Note 183311](#)).

DB Settings

DB Version	SAP Note	Title / Comment
Db2 11 for z/OS	1863848	Installation Parameter Settings for DB2 11
Db2 12 for z/OS	2239553	Installation Parameter Settings for DB2 12

DB Limits

Be aware of the following limits in Db2 11 for z/OS (<https://tinyurl.com/IBM-DB2-zOS-11-1-Limits>). In Db2 12 for z/OS (<https://tinyurl.com/IBM-DB2-zOS-12-1-Limits>) those limits are beyond CDS requirements.

Limit	Db2 11 for z/OS
Maximum table references in one query	225
Number of elements in an IN list	750

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / Topic
2380199	≥ 740	IBM DB2 Analytics Accelerator support with Convergent Invoicing
2432965	750	Enable support for DB2 12 LEFT OUTER TO ONE
2403112	≥ 740	CATMAINT SCHEMA SWITCH and VIEW/UDF hierarchies
2270401	≥ 740	SAP Optimizer profiles for DB2
2385679	≥ 740	Issue with dependent objects during move nametabs
2313326	≥ 740	CDS div() SQLCODE -419
2214151	750	CDS SAP functions LEFT and RIGHT fail to split surrogates
2142546	≥ 740	Rapid replenishment performance

SAP Life-Cycle Management

SAP Note	LC Mgmt. Task or Tool	Title
2403112	System clone	CATMAINT SCHEMA SWITCH and VIEW/UDF hierarchies

Tuning Measures

Transaction	Parameter / Metric to be checked	Activity / Comment
<i>DBA Cockpit → Jobs → DBA planning calendar</i>	To ensure that adequate Db2 catalog statistics are available for a certain CDS query, run the query at least once so that the Db2 12 query engine automatically generates a Runstats profile for the table.	In the Planning Calendar of the DBA Cockpit, schedule <i>Update stats for recommended objects</i>
<i>DBA Cockpit → Statement Cache statistics</i>	I/O wait time	Increase the size of the Db2 buffer pool to minimize I/O wait times
	High number of Getpages	Check if Getpages are due to aggregation or due to an inefficient access path
<i>DBA Cockpit → Performance → Storage → Workfiles</i>	Overflow into physical tablespace or workfile not created because of storage condition	Increase the size of the buffer pool used for workfiles
<i>DBA Cockpit → Performance → Subsystem Activity → RID list processing</i>	RDS limit exceeded	Increase ZPARM MAXRBLK sufficiently to minimize “RDS limit exceeded”; this may require more memory for the z/OS LPAR

More information can be found in Section “Monitoring and Tuning of SAP CDS Queries” of the Db2 DBA guide (https://help.sap.com/viewer/db2_administration_guide).

Microsoft SQL Server

DB Levels

DB Version	SAP Note	Title / Comment
SQL Server 2014	1966701	Setting up Microsoft SQL Server 2014
SQL Server 2016	2201060	Setting up Microsoft SQL Server 2016
SQL Server 2017	2484674	Setting up Microsoft SQL Server 2017

It is highly recommended to use a current version of Microsoft SQL Server plus the latest Cumulative Upgrade (see <https://technet.microsoft.com/en-us/library/ff803383.aspx>).

DB Settings

DB Version	SAP Note	Title / Comment
SQL Server 2014	1986775	Configuration Parameters for SQL Server 2014
SQL Server 2016	2312935	Configuration Parameters for SQL Server 2016
SQL Server 2017	2484657	Configuration Parameters for Microsoft SQL Server 2017

DB Limits

Microsoft SQL Server restricts the number of nested views and functions to 32. You can use the SQL Dependency Analyzer in ABAP in Eclipse to display the SQL Dependency Tree (open *Data Definition in AIE* → *Open With* → *Dependency Analyzer*) to get an overview about the current nesting level of the CDS view.

A list of all SQL Server database limits can be found here:

<https://docs.microsoft.com/en-us/sql/sql-server/maximum-capacity-specifications-for-sql-server>

SAP Notes & Patches for CDS

SAP Note	Component	Comment
2404584	SAP_BASIS ≥ 740	Activation of CDS view fails on MSS
2510415	SAP_APPL 618	CDS Views cannot be created due to database limitation (function nesting level)
2125387	SAP_APPL 617	Activation of CDS view aborts on MSS
2480994	SAP_BASIS 740/750	RS_CDS_COMPARE_DDIC_DB dump REGEX_TOO_COMPLEX
2338548	SAP_BASIS 750	Retcode 512: SQL-error "208-Invalid object name 'k9y.TSTMP_CURRENT_UTCTIMESTAMP'"
2279757	SAP_BASIS 740	RUN_RUTCNVFUNCCRE_BRI: Unknown message with "DROP" "CURRENCY_CONVERSION"
2423050	DBSL ≥ 721	CDS objects are not needed for DBCON remote connections to SQL Server
2087416	DBSL 742	ABAP runtime DBSQL_SQL_ERROR message: Missing end comment mark

SAP Life-Cycle Management

Use SAP Software Provisioning Manager (SWPM) for system copies.



Tuning Measures

Standard SQL Server tuning measures apply. In case of a persistent performance issue, you may also open an incident using component **BC-DB-MSS**.

Oracle Database

DB Levels

Apply the latest available SAP Bundle Patch (SBP):

DB Version	SAP Note	Title / Comment
Oracle 12.1	1888485	Database: Patches for 12.1.0.2
Oracle 12.2	2507228	Database: Patches for 12.2.0.1 (12.2)
Oracle 18c	2660044	Patches for 18c: Database
Oracle 19c	2799920	Patches for 19c: Database

DB Settings

DB Version	SAP Note	Title / Comment
Oracle 12.1	1888485	Database Parameter for 12.1.0.2
Oracle 12.2	2470718	Oracle Database Parameter 12.2/18c/19c
Oracle 18c	2470718	Oracle Database Parameter 12.2/18c/19c
Oracle 19c	2470718	Oracle Database Parameter 12.2/18c/19c

DB Limits

Usage of CDS Views with parameters can cause a performance issue related to a pushdown of predicates with all Oracle Versions earlier than 19.6. A detailed description of the problem is described below. As of Oracle Version 19.6 this DB Limit does not exist anymore as long as the SQL statement text of the CDS Views with parameters does not exceed the length of 32K. For details and necessary prerequisites see SAP Note [2801989](#).

Performance Problems with CDS Views with parameters for Oracle Versions earlier than 19.6:

Usage of CDS Views with parameters can cause a performance issue as a pushdown of predicates is not done by Oracle's execution engine. Therefore, it is recommended not to use CDS Views with parameters in scenarios where a pushdown of predicates is required for performance reasons. This can include filter conditions and join conditions as well. It is planned to fix this issue in one of the upcoming Oracle database versions.

The following simple example describes the problem:

First, a table `tst1` with fields `f1 (number(10))` and `f2 (varchar2(12))` is created with one million rows. 2 rows contain in field `f2` the value 'BBB' whereas the remaining 999998 contain the value 'AAA'. In addition, an index `i_tst1` on field `f2` is created.


```

create table tst1 (f1 number(10) default 0 not null, f2 varchar2(12) default ' ' not null);
begin
  for i in 1 .. 499999 loop
    insert into tst1 values (i, 'AAA');
  end loop;
end;
/
insert into tst1 values (500000, 'BBB');

begin
  for i in 500001 .. 999999 loop
    insert into tst1 values (i, 'AAA');
  end loop;
end;
/

insert into tst1 values (1000000, 'BBB');

commit;

create index i_tst1 on tst1 (f2);

```

Now, a table function (= CDS View with parameters) and a common database view (= CDS View without parameters) are created. Both contain a simple **select** statement on table **tst1** and a **where** condition on field **f1**:

Table function:

```

select * from tst1 where f1 <= tfunc.p_f2 (tfunc.p_f2 is an input parameter of the table
function)

```

Common database View:

```

select * from tst1 where f1 <= 500000;

```

Table Function (= CDS View with Parameters)

```

create type tfunc_l as object (f1 number(10), f2 varchar2(12) );
create type tfunc_t as table of tfunc_l;

create function tfunc (p_f2 numbe)
return tfunc_t pipelined is
begin
  for i in (
    select * from tst1 where f1 <= tfunc.p_f2
  ) loop pipe row( tfunc_l(
    i.f1,
    i.f2 ));
  end loop;
end tfunc;

```

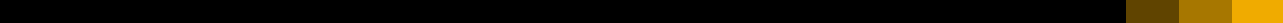
Common Database View (= CDS View):

```

create view tview as
select * from tst1 where f1 <= 500000;

```

In the next step, a select statement on the table function and on the database view is executed. Both statements contain a **where** condition on field **f2** (**f2 = 'BBB'**). The table function has the value **500000** as input parameter. Therefore, both select statements are logically the same and retrieve only one row.



```
select * from table(tfunc( p_f2 => 500000 ))
  where f2 = 'BBB';
```

```
      F1 F2
-----
500000 BBB
```

Elapsed: 00:00:00.45

```
select * from tview
  where f2 = 'BBB';
```

```
      F1 F2
-----
500000 BBB
```

Elapsed: 00:00:00.02

As a result, we can see that the runtime of the `select` statement using the table function is by a factor of more than 20 slower. The reason is, in case of the common database view, that the where condition `f2 = 'BBB'` can be evaluated as a filter at the lowest level in the processing stack. In contradiction to the case where the CDS View with parameters is used, first, all 500000 rows are retrieved by the table function and then the filter `f2 = 'BBB'` is applied, afterwards.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				30 (100)	
* 1	COLLECTION ITERATOR PICKLER FETCH	TFUNC	82	164	29 (0)	00:00:01

Query Block Name / Object Alias (identified by operation id):

1 - SEL\$F5BB74E1 / KOKBF\$0@SEL\$2

Predicate Information (identified by operation id):

1 - filter(VALUE(KOKBF\$)='BBB')

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				449 (100)	
* 1	TABLE ACCESS FULL	TST1	42114	863K	449 (2)	00:00:01

Query Block Name / Object Alias (identified by operation id):

1 - SEL\$1 / TST1@SEL\$1

Predicate Information (identified by operation id):

1 - filter("F1"<=:B1)

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / Comment
2452203	740 – 751	Error in CDS views with parameters for CAST from DEC to CHAR
2347391	740 – 751	Error during activation of CDS views with parameters due to naming conflicts in input parameters
2310324	740 – 750	Error in function SAP_SUBSTRING
2104388	740	Error during activation of CDS views with parameters due to naming conflicts in the database

SAP Life-Cycle Management

No recommendations specific to Oracle.

Tuning Measures

Standard Oracle tuning measures apply. For details refer to:

- <https://tinyurl.com/SAP-NW750-DB-Guide-Oracle>
- <https://tinyurl.com/Oracle-Monitors-and-Tools>

SAP Adaptive Server Enterprise (SAP ASE)

DB Levels

It is highly recommended to use latest available SAP ASE version and patch level.

DB Version	SAP Note	Title / Comment
16.0 SP03	2505861	Release information for SAP ASE 16.0 SP03 with SAP Business Suite

DB Settings

DB Version	SAP Note	Title / Comment
ASE 16.0	1581695	Configuration Guide for SAP ASE 16.0

DB Limits

Refer to <https://tinyurl.com/SAP-ASE-16-Specifications>.

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / Comment
2406419	≥ 740	c-based functions for cds-views
2173215	≥ 740	CDS Views with parameter
2579511	≥ 740	Note collection for CDS with ASE

SAP Life-Cycle Management

SAP Note	LC Mgmt. Task or Tool	Title / Comment
2587717	DB Update	saphostctrl - temporary objects when recreating CDS objects
2438875	SUM	Errors when calling function DATE_IS_VALID
2500400	SUM	Error COMPUTE_BCD_OVERFLOW during SUM Upgrade
2340482	SUM	CDS Views using sap_upper / sap_lower

When running a database update by utilizing SAP-Host-Agent, make sure to use at least PL39.

When executing maintenance activities with Software Update Manager (SUM), make sure to use at least SUM 1.0 SP22 PL12 or SUM 2.0 SP3 PL7.

Tuning Measures

Ensure that regular statistics updates for the complete system are scheduled within the DBA cockpit. More details are available at:

- <https://tinyurl.com/DB-Administration-SAP-on-ASE>
- <https://tinyurl.com/DBA-Guide-SAP-on-ASE>

SAP HANA DB

DB Levels

It is recommended to use latest available SAP HANA DB version and patch level.

DB Version	SAP Note	Title / Comment
1.0	2021789	SAP HANA 1.0 Revision and Maintenance Strategy
2.0	2378962	SAP HANA 2.0 Revision and Maintenance Strategy

DB Settings

SAP Note	Title / Comment
2036111	Configuration parameters for the SAP HANA system

DB Limits

Avoid cyclic **OUTER** joins.

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / Comment
1973233	740	ABAP Dictionary/CDS enhancements and corrections for SAP HANA database
2342494	750	ABAP Dictionary/DB consistency check reports errors for CDS views
2423642	751	Activation error for CDS view with FLTP_TO_DEC conversion
2183911	740	CDS: Incorrect resulting set for view with SUBSTRING function

SAP Life-Cycle Management

For the SAP Software Update Manager, all prerequisites (version, patches, etc.) are checked during the initial phases of the tool.

Tuning Measures

To learn about SAP HANA DB and its administration in general refer to the following links:

- https://help.sap.com/viewer/p/SAP_HANA_PLATFORM
- https://help.sap.com/viewer/p/SAP_HANA_COCKPIT
(covers monitoring, managing, and analyzing performance within the SAP HANA Cockpit)

Particularly helpful in case of database specific problems are the following sources of information:

- <https://wiki.wdf.sap.corp/wiki/display/SuiteCDS/CDS+View+Performance>
- SAP HANA Troubleshooting and Performance Analysis Guide available at:
<https://tinyurl.com/HANA-Performance-Analysis-2-04>.

SAP MaxDB

DB Patch Levels

It is highly recommended to use latest available MaxDB version and patch level.

The minimum requirement is **MaxDB 7.9.09.05**, refer to the following SAP Release Note for details:

SAP Note	Title / Comment
1444241	Feature List for SAP MaxDB 7.9

DB Settings

Set the MaxDB database parameters per SAP recommendations (also see [SAP Note 1346964](#)):

SAP Note	Parameter	Value
2367203	CommandBufferSize	524288

DB Limits

Limit	Value
Number of join tables	254
Number of parameters in a SQL statement	10000

SAP Notes & Patches for CDS

SAP Note	SAP BASIS	Title / topic
2600424	≥ 740	Important information and note collection for CDS with SAP MaxDB
2137905	≥ 740	ABAP CDS: SELECT on CDS view returns sqlcode -8006 and sqlerrtext "Data types must be compatible"
2137955	≥ 740	ABAP CDS: CDS view with CASE statement without ELSE branch
2169157	≥ 740	ABAP CDS: CDS view with RIGHT/LEFT function
2302503	750	MaxDB: Error during generation of CDS views
2310481	≥ 740	ABAP CDS: CDS view with UNION select dumps with error code -9407
2345104	750 – 751	CDS view activation terminates with SQL error -3008 "Invalid keyword or missing delimiter"
2423642	751	Activation error for CDS view with FLTP_TO_DEC conversion
2432946	≥ 740	Error -4023 for creation of CDS view

SAP Life-Cycle Management

No recommendations specific to MaxDB.

Tuning Measures

Ensure that regular statistics updates for the complete system are scheduled within the DBA cockpit.



TRAINING

The table below lists SAP training options related to ABAP CDS.

Training ID	Title
BC430	ABAP Dictionary
S4D430	Building Views in Core Data Services ABAP

APPENDIX

List of changes

February 2020

- Reviewed and updated all web links
- Added Figure 2 (page 6)
- Added minor changes to:
 - Section “Audience” (page 6)
 - Chapter “Best Practices” (page 14ff)
- Updated the following database specific chapter:
 - “IBM Db2 for Linux, Unix and Windows” (page 22): added Db2 LUW 11.5
 - “Oracle Database” (page 32): added Oracle 19c

July 2019

- Updated the following database specific sections:
 - “IBM Db2 for i” – “DB Levels” (page 19)
 - “Microsoft SQL Server” – “DB Levels” / “DB Settings” (pages 30ff)
 - “Oracle Database” – “DB Levels” / “DB Settings” (page 32ff)
- Updated section “Documentation Links” (page 9)
- Reviewed and updated web links

January 2019

- Changed title from “ABAP Core Data Services on anyDB | Best Practice Guide” to “ABAP Core Data Services | SAP Business Suite – Best Practice Guide”
- Added information related to SAP HANA DB throughout the document
- Added section “SAP HANA DB” (page 37) to chapter “Database Specifics”
- Updated section “Development Rules and Governance” (page 15)
- Added section “Performance Safeguarding” (page 16)
- Added SQL Server 2017 to section “Microsoft SQL Server” (page 30)
- Updated SAP ASE specific patch levels of SUM and SAP-Host-Agent in section “SAP Life-Cycle Management” (page 36)
- Added section “Training” (page 3939)
- Reviewed and updated web links

Figures

Figure 1: Principles of ABAP Core Data Services	5
Figure 2: Reduction of communication costs	6
Figure 3: ABAP in Eclipse development framework	9
Figure 4: Components and activities when exposing CDS views to OData	10
Figure 5: CDS consumption by SAP Fiori - architecture overview	11
Figure 6: Analysis for Microsoft Office with Excel and PowerPoint	11
Figure 7: SAP Convergent Invoicing	12




Figure 8: Db2 LUW Explain within the DBA Cockpit	23
Figure 9: Db2 LUW Access Plan Graph from IBM Data Server Manager	24
Figure 10: Db2 LUW output of SQL listed in Figure 11	25
Figure 11: Db2 LUW SQL statement to retrieve an object hierarchy	26
Figure 12: Db2 LUW dependencies and references for an object in the IBM Data Server Manager	27

References

SAP Roadmaps

<https://www.sap.com/products/roadmaps.html>

SAP Community

<https://go.sap.com/community.html>

SAP Products

<https://www.sap.com/products.html>

ABAP Development

<https://www.sap.com/community/topics/abap.html>

SAP Fiori / SAP User Experience

<https://www.sap.com/products/fiori.html>