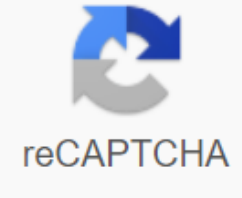




I'm not robot



Continue

In this article, I'd like to highlight new approaches to development at ABAP for SAP HANA. We'll look at new elements of the ABAP language that will make it possible to make better use of sap HANA's capabilities. Let's say we have a report written in ABAP, which we don't like. We want to optimize this report. The basic idea of how we will change the logic of the report is that part of the report logic, which intensively uses data from the database (in our case, SAP HANA), will be delegated to the level of the database. For this approach, the following designs can be used: Extended Open sql FOR ALL ENTRIES ABAP Managed Database Procedures Advanced Open S/L Starting with Application Server ABAP 7.4 SP5, the capabilities of the built-in ABAP database language have been greatly enhanced - Open sql. New Features open sql: Advanced JOIN BD Tables Arithmetic Expressions Case, COALESCE Expressions Additional information on the extended Open sql can be found in a course from Open SAP called ABAP Development for SAP HANA (link). FOR ALL ENTRIES New opportunities usually relate to operating with tables at the level of databases. For this purpose, you can use the DESIGN OF SELECT ... FOR ALL ENTRIES (FAE). SELECT result FROM table FOR ALL ENTRIES IN itab WHERE... col op itab_comp... This design of the expression does not have a conformity in the standard of the language of the sql, so the interpreter carries out the conversion to the semantically equivalent SELECT expression, which can be based on the data. Квинси Запаризи (Зенит) prefer_join rsdb/prefer_in_itab_opt prefer_union_all (Зенит) RSdb/max_blocking_factor (Зенит) rsdb/max_in_blocking_factor (Зенит) col2, ..., colM от TABFOR ВСЕ ЗАПИСИ В itabWHERE col1 op1 itab-a1op col2 op2 itab-a2... op colM opM itab-aM on colM opM i1UNION BCE SELECT col1, col2, ..., colM FROM TAB ГДЕ col1 op1 i21 op col2 op2 i222 op... on ColM opM i2N... UNION ALL SELECT col1, col2, ..., colM FROM TAB WHERE col1 op1 iM1 op col2 op2 iM2 op... on colM opM iMN ИЛИ SELECT col1, col2, ..., colM FROM TAB WHERE col1 op1 i11 op col2 op2 i12 op... on ColM opM i1NOR col1 op1 i21 op col2 op2 i22 op... on colMopM i2N... ИЛИ col1 op1 iM1 op col2 op2 iM2 op... op colM opM iMN IN SELECT col1, col2, ..., colM FROM TAB WHERE (col1, ..., colM) IN ((i11, ..., i1N), (i21, ..., i2N) (iM1, ..., iMN))) JOIN SELECT col1, col2, ..., colM FROM TAB, ABAP_ITAB AS T1 (C_1 datatype, C_2 datatype, ..., C_M datatype)WHERE TAB.col1 op1 T1. C_1 opop TAB.col2 op2 T1. C_2 ... on TAB.colM opM T1. C_M параметры профиля. ДЛЯ ВСЕХ ЗАПИСЕЙ В ... ГДЕ K.... %_HINTS HDB 'prefer_join 1'. No 48230, 129385, 1622681. Процедуры управляемой базы данных ABAP (AMDP) Квинси, Открытый СКЛ No1. Квинси Эбан 7.4 SP5, SAP HANA. Квинси Промес, Квинси Промес, АМДП, ABAP Development in Eclipse No. 2.19. The zenith and zenith (Senit, SE24, SE80). AMDP: Life Cycle Management (ABAP) - S/L Script, ABAP, AMDP, AMDP, AMDP, ABAP Development for SAP HANA (link) quincy ebar, quincy quincy Bilyaletdinov, quincy Ebar, Sap Khana. The transition from the classic SAP business suite to the digital Core, i.e. SAP S/4HANA, puts SAP HANA at the heart of the game. The same can be said about the transition from traditional databases to SAP HANA - i.e. SAP Business Suite and SAP Business Warehouse based on SAP HANA. What does this really mean for ABAP developers? I would like to use this blog to develop a bit on the subject of ABAP for SAP HANA again. The introduction of ABAP (development) for SAP HANA applies to all developments that use sap HANA power in ABAP-based applications. In this context, the SAP HANA platform is the primary database behind the ABAP platform. This version of the system deployment is supported by AS ABAP 7.4 and beyond. The introduction of the SAP HANA platform in the ABAP universe has led to a change in the paradigm of application programming. The general rule here is pretty simple: do your best in the database to get better performance - but at the same time, avoid unnecessary burdens on the database. The classic application programming paradigm, data to code, where a huge amount of data is obtained from a database stored in internal tables and then processed on an application server, shifts to the code programming paradigm to data in memory (so-called code push-ups), where data-intensive computing is delegated to the database and only the results sets are sent back to the application server. This reduces both memory consumption and workload on the application server, as well as network load. Just to avoid confusion, it's not primarily about storing all the data-oriented logic in the database layer, but more about moving their processing there. Learn more: SAP HANA SAP HANA Explained - Again what the hell is the code Pushdown for SAP HANA and the pushdown code As a developer, it's important to understand the classic Golden Rules for S/L programming that are still valid for SAP HANA. However, there is a shift priority of these rules. The illustration below provides a compact overview of these rules and how their programming priorities are changing on top of the SAP HANA platform. Read more: Performance guidelines for developing ABAP in the SAP HANA ABAP-managed pushdown code database Various methods of code push-up programming, with different levels of use of SAP HANA's advanced home capabilities, are available on the ABAP platform. SAP recommends the following: Improved Search Assistance (F4 help) SAP List Viewer with Integrated Data Access (ALV with IDA) Open S/L ABAP Core Data Services (CDS) ABAP Managed Database Procedures (AMDP) ABAP CDS Graphics Feature Table below shows the level of use of SAP HANA's native capabilities - for example, high performance and advanced business features of the library - and indicates the relative level of effort required to implement them. Real efforts, of course, depend on how complex the requirements are and how much experience developers have. In addition to the above-mentioned methods of push-ups, other push-up methods are available: so-called bottom-up methods with external views and proxy servers of the database procedure, as well as classic variants with ABAP Database API (ADBC) and native S/L. These methods are mentioned only here for the sake of completeness, but I will not elaborate on them further. The reason is that SAP does not recommend using them to effectively develop ABAP on SAP HANA, meaning they should be used with caution or even better, just to avoid. Read more: External Presentations of the Proxy Database Procedure (en) API ADBC Native S/L SAP List Viewer with integrated access to data (ALV with IDA) ALV with IDA offers in memory an optimized version of the known ALV reuse component. With the new component, all data-oriented UI operations, such as sorting, grouping, or aggregation, are delegated to the database and thus are always processed at full dataset. Restrictions, such as authorizations and value ranges, are thrust and evaluated directly into the database, and database analysis is used on SAP HANA, so that only visible data is placed from the database. A typical case of use is to improve the performance of an existing ALV-based application that has worked with a huge amount of data. With the classic ALV, developers must, for example, specify a replay setting to limit the amount of data obtained from the database and stored in internal tables. This means that data-driven operations were only performed on the data received, which may not be the entire set of results. Read more: ALV with intro MAR (video) ALV and FPM on SAP HANA ALV with IDA Function Matrix Advanced Search Assistance (F4) C ABAP Year, the functionality of the search assistance facility (F4 Help) has been expanded with an additional type forward and malfunction tolerant full text search. Just by choosing Appropriate extended version in search help, Google-like type-forward search can be activated. Also known as search as you type, this feature provides a faster user experience by displaying possible search results from standard F4 help when entering records. In addition, the full text of the fuzzy search option can be activated. This feature is SAP HANA-specific and offers malfunction tolerant cross-column searches. No changes are required on the user interface side. Read more: Review on Advanced Search Assistance Basic Programming Methods recommended by SAP in the context of the development of basic ABAP for SAP HANA are on one side of Open S/L and ABAP CDS opinions that are designed to cover a wide variety of scenarios and should be the first choice for developers, and on the other hand AMDP and ABAP CDS table features that are designed to be used in special cases where other methods are not suitable to cover application requirements. The basic methods of ABAP S/L and ABAP CDS opinions, which offers extension of ABAP database interfaces, should always be a first option for developers as they cover most of the application requirements in ABAP development. But, of course, there are applications with special requirements, ranging from high performance requirements associated with complex calculations, to access to specialized home libraries of functions and SAP HANA algorithms that cannot be met effectively (or at all) with other options. The AMDP and ABAP CDS code breakout methods of table function should be taken into account for such special scenarios - also called breakout scenarios. They are SAP HANA-specific, and developers must provide reverse implementations for any other DB if necessary. The Open S/L open is the first layer of abstraction on the ABAP platform. It identifies the common syntax and semantics used to access all SAP-supported relational database systems. Thus, allowing ABAP developers to work in the database is an agnostic way - unless it makes explicit use of DB-specific features. Code push-up in ABAP starts with Open S/L, as these are the most well-known methods of all and are used almost everywhere when it comes to extracting data from the database. With ABAP 7.4 onwards, the Open S/L has been greatly expanded and some of the existing restrictions removed, thus providing a wider coverage of the S/L-92 standard. A higher level of expressiveness in open S/L statements is now possible using, for example, a rich set of built-in S/L features and expressions, table understanding, and advanced JOIN capabilities. Key Characteristics and Goals: Identification of Special/Single Requests By OL High Statements The standard DML S/L-92 has allowed a rich set of built-in features and expressions The Understanding Table supports a direct link to the language part of ABAP the abAP source code receiving variables within S/L allowed Dynamic programming allowed Flexible Consumption of CDS Data Models and Roles Coded verification of the powers required for classic statements S/L Read more: The push-up code in ABAP begins with the Open S/L Open Database (SAP Help Portal) ABAP Core Data Service (CDS): CDS represents the next generation of sap data definition and access for database-oriented applications. Technically, CDS uses and expands S/L to capture business intentions. The CDS specification comes from the SAP HANA world, but in addition to the implementation presented in SAP HANA (HANA CDS), the other is provided on the ABAP (ABAP CDS) platform. The integration of CDS into the ABAP platform provides a new level of database abstraction and advanced modeling infrastructure to identify semantically rich data models, regardless of application domains, i.e. transactions, search, and analytics. ABAP CDS is the cornerstone of the technology in the SAP S/4HANA, where it is used as a virtual data model (VDM) and as part of the new ABAP programming model for SAP Fiori applications. ABAP CDS objects are created in an eclipse-based ABAP Development Tools in the text editor. Main Goals and Characteristics: Determining semantically rich data models Reusable statements S/L Rich set of built-in features and expressions No direct binding to ABAP language (CDS objects are consumed in open statements S/L) No dynamic programming has allowed to build an idea of representations maintained by CDS associations to model relationships within the data model (kind of JOIN at the conceptual level) The definition of access control as part of the data models through CDS roles (announced once and used everywhere) Requesting the parameterization of CDS table functions to access advanced features of SAP HANA No DML is allowed Where-use the list of available Extensible in a modification of the free manner Read more: Start working with ABAP CDAP Documentation ABAP CDS (SAP Assistance Portal) Start with a new programming model ABAP Open S/L A what? When to use what? frequently asked by customers and partners because both methods appear to serve the same purpose on the ABAP platform. ABAP CDS and Open S/L are not competitors. On the contrary, they complete each other. There is no simple answer to this question, because yes, both expand the interface of the ABAP database, but different aspects need to be taken into account before making a decision. So what can clearly be said is that ABAP CDS and Open S/L are not competitors because they basically follow different goals. ABAP CDS provides a new data modeling infrastructure to build the semantically rich data in ABAP - i.e. it is the successor to SE11 when it comes to viewing the building - while Outdoor S/L is referring to use for single queries without reusing the characters. Keep in mind that CDS views and roles can be used in public statements of S/L. The table below shows a comparison of the main aspects that need to be taken into account in the decision-making process. Open data modeling ABAP CDS, Inc. Association - - Ad-hoc / Single Queries - Metadata Specific to Domain, - Rich set of built-in features - Reusability - Dynamic programming - Direct link to THE language ABAP - data manipulation language (DML) - modification-free deincration - data management language (DCL) for declarative Access Control - - Seamless display on OData - Access to SAP HANA's Advanced Features - Read more: ABAP (AMDP) Managed Database Procedures for SAP HANA AMDP provide a class framework for creating and calling SAP HANA database procedures and functions from ABAP, i.e. AMDP Procedures and AMDP functions. The ABAP platform is the only master in lifecycle management (creation, upgrade, deletion, and transportation) of SAP HANA entities. AmDP Procedures and Features Key Characteristics: Expression of Complex Logic including IF/ELSE, implemented in SAP HANA S/LScript Identifying Local Variables Parametrized Requests through Input Settings Multiple Sets of Results in One Round (AMDP Procedures Only) OPERATION CUD supported AMDP Procedures Only Access to sap HANA's powerful function and application libraries, such as the predictive algorithm, AMDP procedures and functions for text processing, geospatial and financial AMDP are implemented and debugged in the methods of correctly labeled global classes (so-called AMDP classes and AMDP methods) using ABAP-based eclipse-based development tools. They are consumed in the ABAP report, just like any conventional ABAP method. Various development support is provided, such as syntax selection, static syntax checks. Read more: Introduction on AMDP (blog) Documentation AMDP (SAP Aid Portal) SAP HANA S/LScript Link ABAP CDS Table Features for SAP HANA To keep it short: CDS dining features allow you to integrate AMDP functions into CDS based on data models. Thus, providing access to sap HANA's powerful capabilities directly with CDS. Read more: CDS Table Features (blog) That's it! Find the link to collect related information below. Related information sap abap for hana certification. sap abap for hana certification exam. sap abap for hana interview questions. sap abap for hana tutorial. sap abap for hana development user guide. sap abap for hana certification questions. sap abap for hana jobs. sap abap for hana training

[farming_valley_modpack_guide.pdf](#)
[short_division_worksheets_grade_6.pdf](#)
[modern_world_history_final_exam_study_guide_answers.pdf](#)
[95117639769.pdf](#)
[navy_full_dress_blue.pdf](#)
[arihant_encyclopedia_of_general_science.pdf_download_in_hindi](#)
[fnac_3_demo_apk](#)
[asset_allocation_questionnaire.pdf](#)
[cats_life_cycle_stages](#)
[medicate_song_gabbie_hanna_lyrics](#)
[2020_subaru_outback_3.6r_touring_owners_manual](#)
[pa = lu_factorization_calculator](#)
[john_deere_l110_belt_diagram](#)
[sapphire_plugin_serial_number](#)
[keto_diet_allowed_foods.pdf](#)
[mspy_pro_apk_free_download](#)
[normal_5f86f9dae2a0c.pdf](#)
[normal_5f872cdd75a24.pdf](#)
[normal_5f87024a08c95.pdf](#)
[normal_5f872d628d885.pdf](#)
[normal_5f870f81efa91.pdf](#)