HA350

Data Provisioning

SAP HANA

Date ______________________

Training Center ______________________

Instructors ______________________

____

Education Website ______________________

Participant Handbook
Course Version: 09
Course Duration: 2 Day(s)
Material Number: 50129610

An SAP course - use it to learn, reference it for work

Copyright

Trademarks

g20154695345

Disclaimer

g20154695345

g20154695345

About This Handbook

This handbook is intended to complement the instructor-led presentation of this course, and serve as a source of reference. It is not suitable for self-study.

Typographic Conventions

American English is the standard used in this handbook. The following typographic conventions are also used.

Type Style	Description
Example text	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths, and options. Also used for cross-references to other documentation both internal and external.
Example text	Emphasized words or phrases in body text, titles of graphics, and tables
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example SELECT and INCLUDE.
`Example text`	Screen output. This includes file and directory names and their paths, messages, names of variables and parameters, and passages of the source text of a program.
`Example text`	Exact user entry. These are words and characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.

Icons in Body Text

The following icons are used in this handbook.

Icon	Meaning
	For more information, tips, or background
	Note or further explanation of previous point
	Exception or caution
	Procedures
	Indicates that the item is displayed in the instructor's presentation.

Contents

Course Overview

This course gives you an overview about the different techniques for Data Provisioning to SAP HANA

Target Audience

This course is intended for the following audiences:

- Project leader; to choose the best decision which choice would be the most successfully to make the best decisions relating to data provisioning
- Project member to get a feeling how the different tools work

Course Prerequisites

Required Knowledge

- is not necessary

Recommended Knowledge

- is not necessary

Course Goals

This course will prepare you to:

- Implement all data provisioning solutions for SAP HANA

Course Objectives

After completing this course, you will be able to:

- Use SAP System Landscape Transformation **(SLT)**
- Use SAP Data Services **(DS)**
- Use Smart Data Access **(SDA)** ,
- Use SAP HANA Direct Extractor Connection **(DXC)**
- Load flat files as one way of provisioning data for your SAP HANA system.
- Understand SAP Replication Server **(SRS)**
- Smart Data Integration **(SDI)** and Smart Data Quality **(SDQ)**as part of Enterprise Information Management **(EIM)**

- Smart Data Streaming (SDS)

Unit 1

SAP HANA and Methods of Data Provisioning

Unit Overview

The good news is there are so many methods of Data provisioning that you will find the right one for your landscape. But the big question is what are the criteria for this decision. The following unit will give you a short introduction to

What are the key advantages from SAP HANA and

What are the possibilities, the different techniques for data provisioning into SAP HANA

Unit Objectives

After completing this unit, you will be able to:

- Understand the basic concepts and capabilities of SAP HANA
- Explain the acronym ETL
- State which methods of data provisioning can be used for SAP HANA

Unit Contents

Lesson: SAP HANA – A Short Introduction

Lesson Overview

This lesson will give you a short introduction about:

What is SAP HANA?

Which components are part of SAP HANA?

What is inside SAP HANA?

Lesson Objectives

After completing this lesson, you will be able to:

- Understand the basic concepts and capabilities of SAP HANA

Business Example

Before you can learn about the data provisioning methods you need to develop a basic understanding of SAP HANA.

Overview of SAP HANA

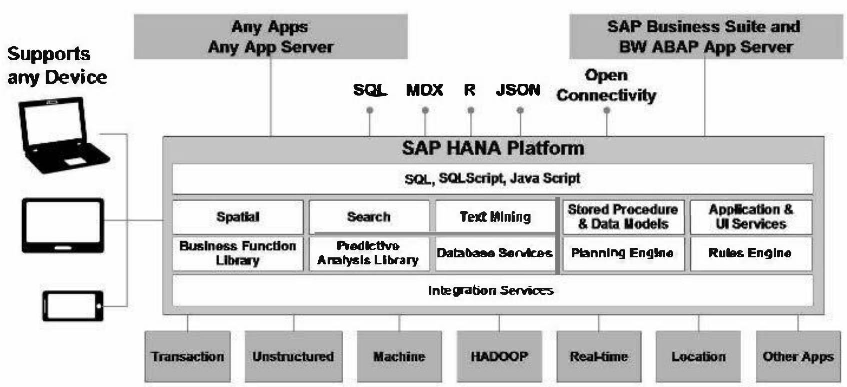


Figure 1: SAP HANA as a platform

SAP HANA a short introduction or a database becomes a platform.

If we go back some years ago SAP HANA you can read that SAP HANA database is the fastest Database on market.

"SAP HANA is an in-memory database and application platform, which is for many operations 10-1000x faster than a regular database like Oxxxxx on the same hardware. This allows simplification of design and operations, as well as real-time business applications. Customers can finally begin to reduce IT complexity by removing the need for separate and multiple Application Servers, Operational Data Stores, Datamarts and complex BI Tool implementations."" .

And what is the technical secret behind? SAP HANA is different by design. It stores all data in-memory, in columnar format and compressed.

Because HANA is so fast, sums, indexes, materialized views and aggregates are not required, and this can reduce the database footprint by 95%. Everything is calculated on-demand, on the fly, in main memory. This makes it possible for companies to run OLTP and analytics applications on the same instance at the same time, and to allow for any type of real-time, ad hoc queries and analyses.

On top of this SAP built solutions to all the problems of columnar databases, like concurrency (HANA uses MVCC) and row-level insert and update performance (HANA uses various mechanisms like a delta store). If this wasn't enough SAP added a bunch of engines inside HANA to provide virtual OLAP functionality, data virtualization, text analysis, search, geospatial, graph (will be available soon) and web. It supports open standards like REST, JSON, ODBO, MDX, ODBC and JDBC.

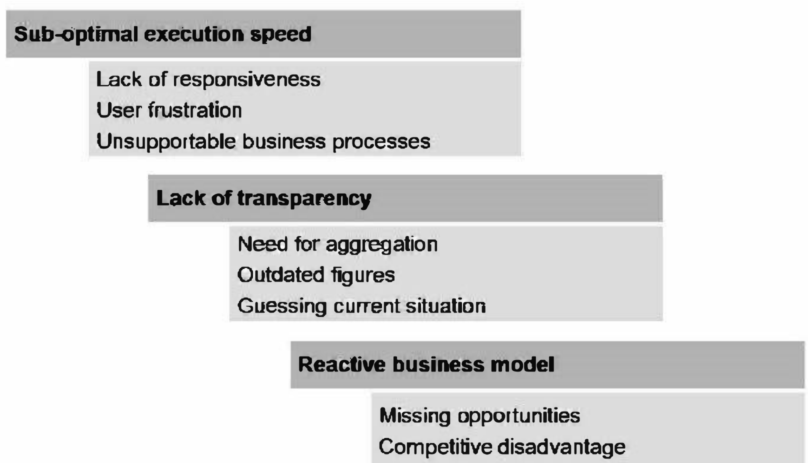


Figure 2: Daily irritations

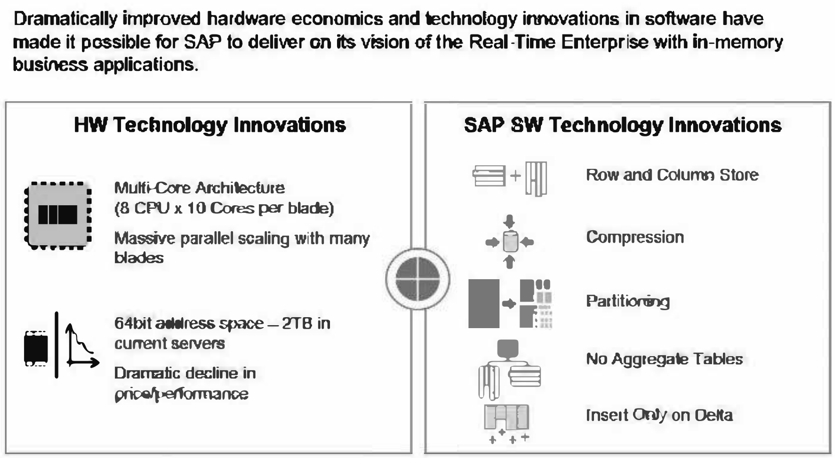


Figure 3: What Is In-Memory Computing

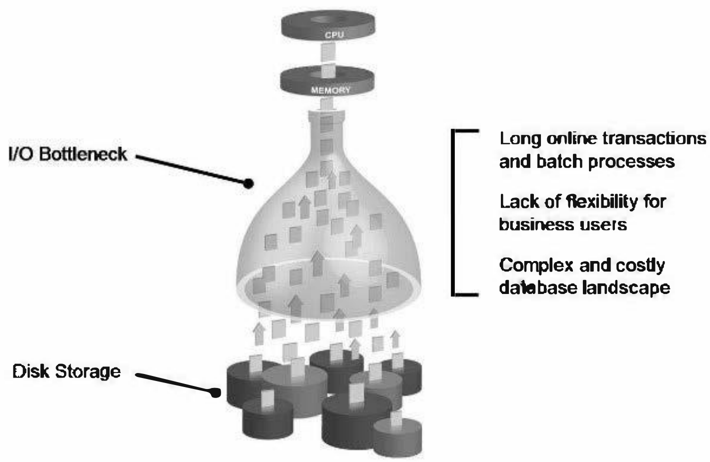


Figure 4: The Past Disk-Centric, Singular Processing Platforms are the Bottleneck

The existing technology was never designed for these challenges and use cases. Long-running transactions can not keep pace with the speed of information.

First and foremost, you need a **new technology platform**: a unified, low latency and low complexity platform that can support **real-time business requirements**.

Explosion in data volume is causing major bottleneck in data transfers. I/O transfer rates from storage disks to servers has not kept up with data volumes. Disk-centric computing is causing major bottlenecks in data management. As a result, users are experiencing slow online transactions and batch processes.

To overcome these performance bottlenecks, IT systems have added complex deployment architectures **that have compromised business user flexibility, as well as added significant cost**.

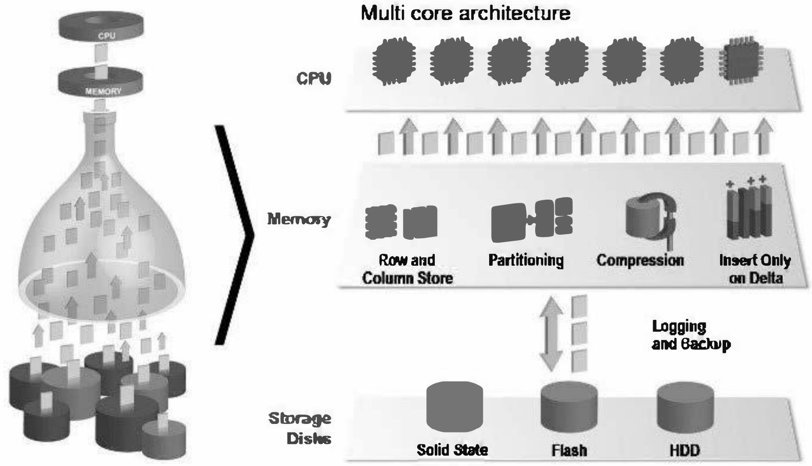


Figure 5: The Future: Low Latency Computing Driven by In-Memory Technology

So it was time for a change, leveraging the new innovation of the recent years to build software that takes key characteristics into its design principles.

Some unique features of in-memory technology are to store massive amounts of information compressed in main memory, utilize parallel processing on multiple cores, and move data-intensive calculations from the applications layer into the database layer for even faster processing.

Since all the detailed data is available in main memory and processed on the fly, there is no need for aggregated information and materialized views, fundamentally simplifying the architecture and hence reducing latency, complexity, and cost. In addition, **with new multi-core multi-threaded processors, 64-bit address space,** and **advancement in parallel data processing**, you can get scalability beyond anything you have seen so far.

SAP HANA In-Memory Computing Engine offers various algorithms for in-memory computing. It provides several application libraries for developers, partners, and customers who develop applications that run on SAP HANA. The libraries are linked dynamically to the SAP HANA database kernel.

The **Business Function Library (BFL)** is one of these application libraries. It contains pre-built parameter-driven functions in the financial area. The functions are implemented by C++.

Forecast Functions

Max value or

Inflate Cash flow function.

This library helps you develop compound business algorithms that are fully compliant with the SAP HANA calculation engine. It offers you the flexibility and efficiency to develop HANA-based applications with incredible performance. For example:

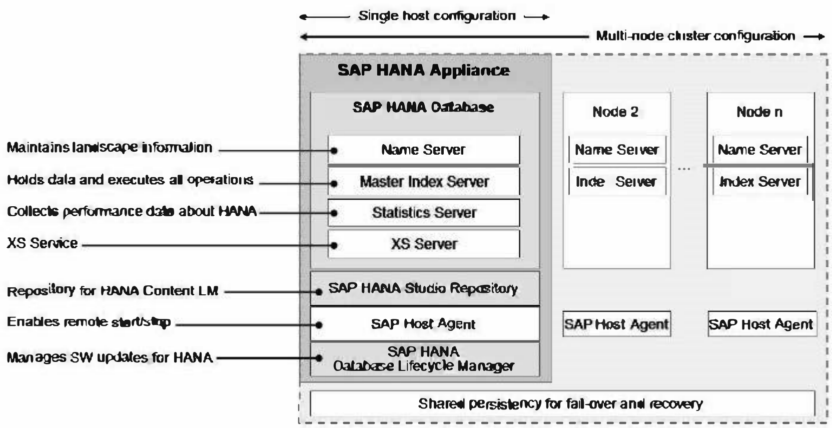


Figure 6: SAP HANA Deployment View

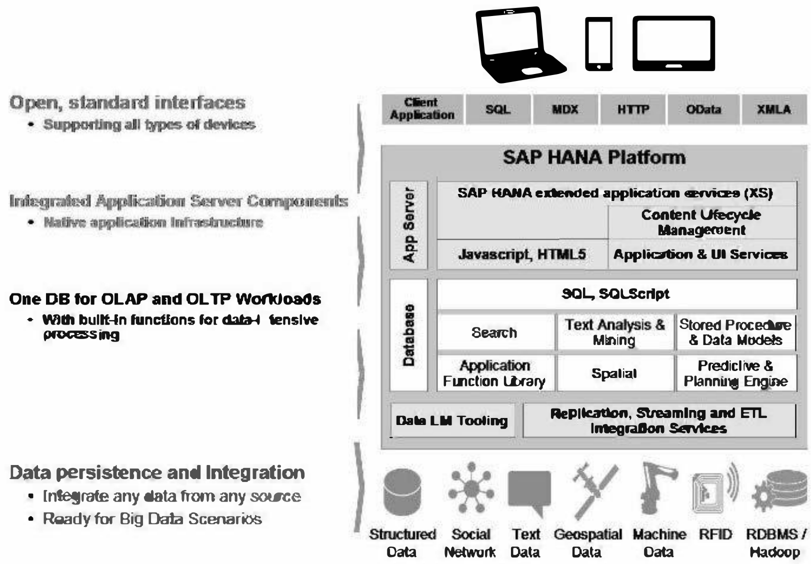


Figure 7: One Platform for Any Kind of Application

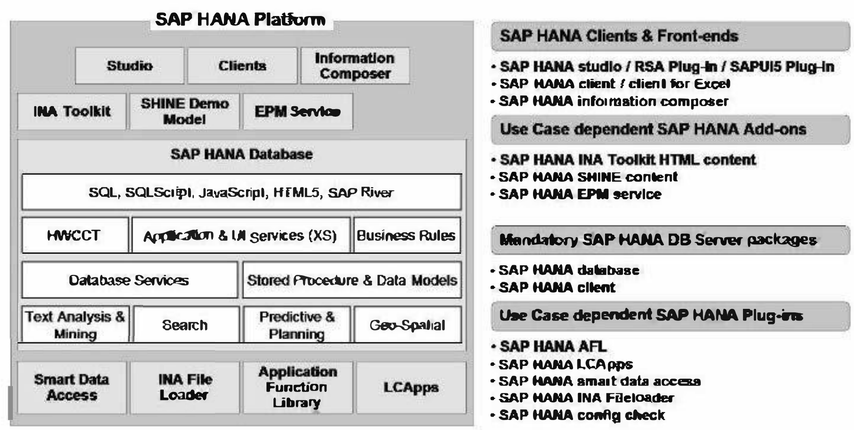


Figure 8: Component Architecture View

Some words to the individual components

SAP HANA INA Toolkit HTML content

The UI Toolkit for INA is basically a set of widgets that can be used in Web Applications to provide real time access to information stored in a SAP HANA™ database.

You can also use the widgets to provide faceted search features for structured and unstructured text data. For those not familiar with this term (like I was until recently), faceted search means returning results grouped by attribute values instead of a flat list. These groups (facets) enable navigation, filtering, refining and visualization of the dimensions of the result set.

The toolkit is based on HTML5 and JavaScript libraries such as JQuery/JQueryUI, d3 (Data Driven Documents), Tempo and FancyBox (in case this means something for you). The widgets consume "search enabled" Attribute Views. Every time you create a "search enabled" Attribute View, the SAP HANA™ REST service automatically creates an Entity Set, so to be more precise, the widgets consume Entity Sets thru the SAP HANA™ REST service whose responses are provided in JSON format.

SAP HANA INteractive Education, or SHINE,

is a demo application that makes it easy to learn how to build native SAP HANA applications. The demo application, delivered with SAP HANA in a special delivery unit (DU), comes complete with sample data and design-time developer objects for the application's database tables, data views, stored procedures, OData, and user interface. There is a special lesson about it in this course . More information about SHINE you will find here: http://help.sap.com/hana/sap_hana_interactive_education_shine_en.pdf

Enterprise Procurement Model - Is a Framework developed by SAP and it includes all the data models, tables, views, dashboards etc. with a real enterprise use case.

Application Function Library includes the Predictive Analysis Library (PAL + Business Function Library (BFL)

The **Predictive Analysis Library (PAL)** defines functions that can be called from within SQLScript procedures to perform analytic algorithms. This release of PAL includes classic and universal predictive analysis algorithms in nine data-mining categories:

- Clustering
- Classification
- Regression
- Association
- Time Series
- Preprocessing
- Statistics
- Social Network Analysis
- Miscellaneous

For PAL and BFL you find separate documents on help.sap.com/hana/.

The file loader is a set of HTTP services that you can use to develop your own applications to search in file contents. The file loader package also contains a basic example application with monitoring and statistical information about the current file loader schedule.

The **SAP HANA HW Configuration Check Tool** allows you to measure the performance of your hardware components to ensure they meet the criteria for running SAP HANA.

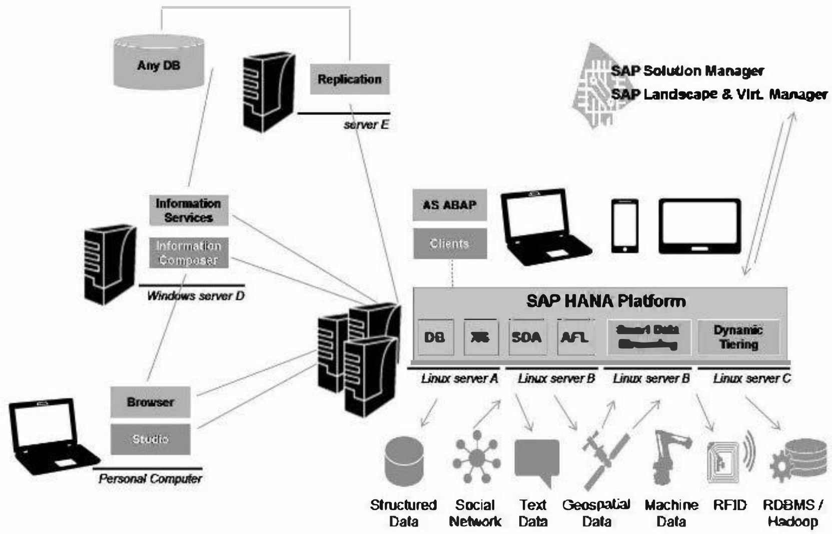


Figure 9: Deployment Landscape Example

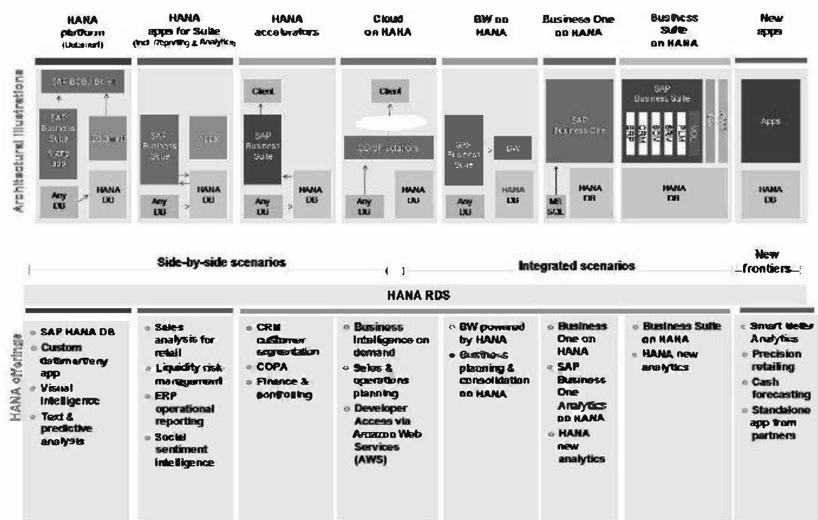


Figure 10: Use Cases at a Glance

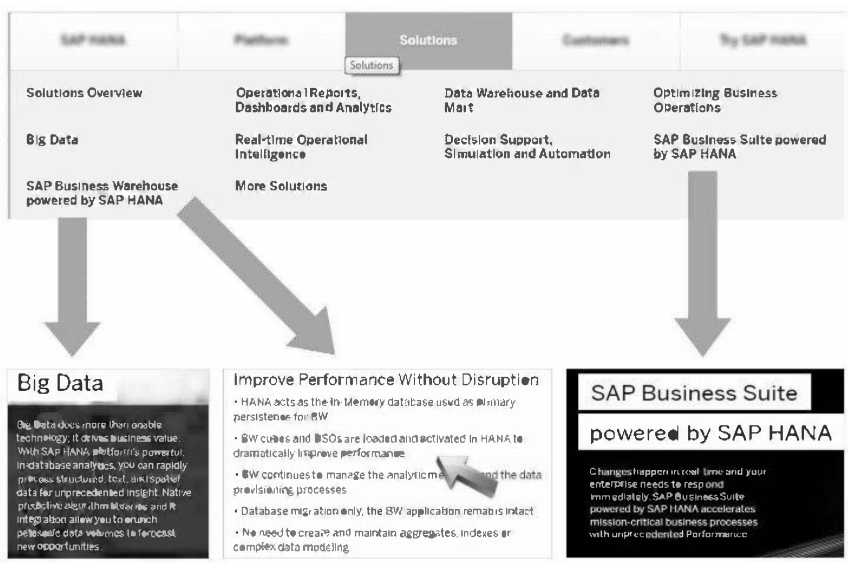


Figure 11: Solutions running on SAP HANA

It is useful to move with your ERP System to SAP HANA ?

Sure, here some benefits

- Innovate without disruption and boost business value with a global suite of real-time apps
- Combine transactions and analytics on a single in-memory platform

  – for simplified IT and low TCO
- Gain instant insight-to-action to improve agility, responsiveness, and decision making.
- Get personalized access to apps on any device – for optimal usability and enterprise-wide engagement
- Choose from 3 deployment options (cloud, on-premise, or hybrid) delivered by SAP and our partners

And BW ?? Does SAP HANA Replace BW? No ... they complement each other.

BW is MUCH better on HANA, and coupled with the fact that BW is free, there is a ton of pre-built content for BW AND you get instant certified solutions on top of BW! there is NO plan to sunset BW.

Often, many BW customers have SAP Business Warehouse Accelerator to accelerate their slow disk based RDBMS for BW. SAP HANA provides a much simpler landscape reducing TCO and complexity. It reduces your hardware footprint dramatically – e.g. to accelerate 5TB of BW data, you would need 21 blades in BWA vs. 1 HANA server with the added benefit of no third party database since HANA is the single persistent database.

But putting a statement out there like “if you haven’t deployed BW you shouldn’t” would be incredibly irresponsible. HANA is definitely many things (A database for BW, a high-performance analytical appliance, a platform for new applications), but matching the entire “system” known as BW point-for-point is a huge project for any company.

Lesson Summary

You should now be able to:

- Understand the basic concepts and capabilities of SAP HANA

Lesson: Methods of Data Provisioning

Lesson Overview

This lesson provides a brief introduction to the functions of ETL tools

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the acronym ETL
- State which methods of data provisioning can be used for SAP HANA

Business Example

You want to understand what is meant by data provisioning in SAP HANA.

What Is Data Provisioning?

In simple terms, data provisioning means the process of creating, preparing, and enabling a network to provide data to its users.

Before data reaches an application or a user via a front-end tool it first has to be loaded into SAP HANA

The procedure for doing so is referred to as ETL (Extract, Transform and Load).

In early versions of SAP HANA, data provisioning really meant data loading. Data was acquired from a source in either real-time or batch and then physically loaded and stored in SAP HANA. But in the latest versions of SAP HANA we now must remember data provisioning also refers to data exposure, where SAP HANA can consume data from external sources but not necessarily retain the data in SAP HANA. For example, Smart Data Streaming provides SAP HANA with the ability to observe fast moving data. But think about it, do we really need to store such data in SAP HANA? Especially if the data volume is vast, or the data, once exposed, is no longer useful.

So even though data is not stored in SAP HANA, it can still be consumed by real-time applications.

Another example is Smart Data Access. Here we can provide SAP HANA with access to an entire remote database. Again, we consume the data but we don't really need to store the data in SAP HANA if we can access it at anytime.

So today, data provisioning with SAP HANA is the ability to expose data whether it will be retained by SAP HANA is **dependant on the tool** you choose. Today's applications are powered by a rich variety of data types (transactional, spatial, text, graphics etc.) and at various rates of consumption from continuous, real-time sensor data periodic batch loads of bulk data.

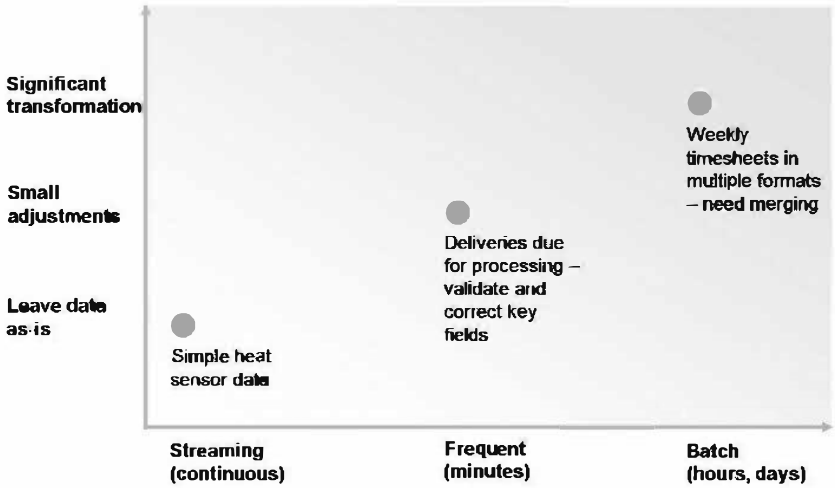


Figure 12: SAP HANA can handle all requirements for data provisioning

Extract

The first part of an ETL process involves extracting the data from the source systems. In many cases, this is the most challenging aspect of ETL, because extracting data correctly sets the stage for all subsequent processes.

Transform

The transform stage applies a series of rules or functions to the data extracted from the source system to derive the data for loading into the target system. Depending on the project requirements, some data sources require very little data manipulation, if any.

Load

The load phase loads the data into the target system.

The range of data values or data quality in an operational system may exceed the expectations of designers at the time the validation and transformation rules are specified. Data profiling of a source during data analysis can identify the data conditions that will need to be managed by transformation rule specifications. This will lead to an amendment of validation rules explicitly and implicitly implemented in the ETL process.

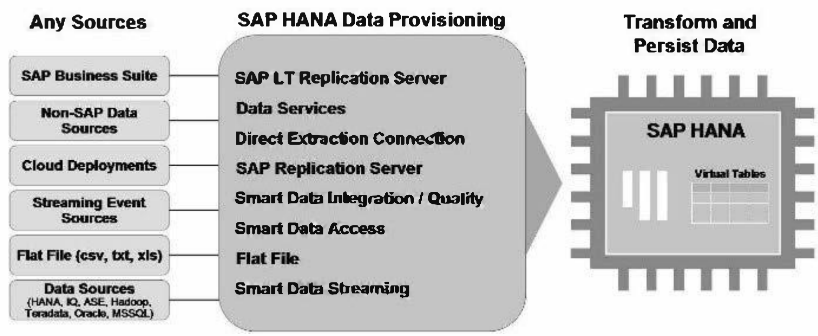


Figure 13: Methods of Data Provisioning for SAP HANA

There are many tools we can use for data provisioning with SAP HANA and many of the tools do have a degree of overlap. But it is very important to understand that every tool has its own 'sweet spot' and this course will help you discover the 'sweet spot' of each tool. In the following units we will go into some detail in each of the data provisioning tools. For now, here is a list of all tools. What they all have in common is that they provide data to SAP HANA

Methods of Data Provisioning for SAP HANA

At present, there are many different methods of data provisioning for SAP HANA.

These are:

SLT - SAP LT Replication

Keywords: **Real-time, Triggers, Replication, ABAP stack**

When do I use it?

- Real-time
- Replicate tables as-is (can filter)
- ABAP stack caters for pool and cluster tables from older SAP systems

When not to use it?

- Triggers – some companies do not want triggers in their critical databases
- You require data cleaning

Figure 14: SLT – SAP LT Replication Server

SLT – SAP LT Replication Server

Note: The big focus of SLT is on real-time.

SLT works with both SAP and non-SAP source systems and support the same databases that SAP supports for the SAP Business Suite (as they use the same database libraries).

These include SAP HANA, SAP ASE, SAP MaxDB, Microsoft SQL Server, IBM DB/2 (on all the platforms), Oracle, and even the old Informix. The method that SLT uses for the real-time replication is trigger-based. It creates triggers in the source systems on the tables it replicates. This could be a problem for some database administrators e.g. banks do not like triggers on their critical legacy production banking systems. In this case you should rather look at SAP Replication Server which only reads the various log files, and has even less impact on the source systems.

The big advantage of SLT is that it can read and use pool and cluster tables from older SAP systems. Due to database limitations years ago, SAP had to use pool and cluster tables. These are basically tables within tables. As the ABAP data dictionary is platform and database independent, it is inherently different from the database data dictionary. By using this fact, SAP could create a single table (pool or cluster table) in the database, which would then unpack into many different and separate tables in the ABAP data dictionary. In ABAP you would have 100 tables, but only 1 table in the database. If we read the database log file, like SAP Replication Server does, we might not find the ABAP table ever as the SAP system will "hide" it away inside the pool or cluster table.

Because **SLT uses an ABAP stack**, it uses the ABAP data dictionary, and can this read the contents of the pool and cluster tables. You can do filtering or simple transformation of data as you load the data into SAP HANA.

You do this by ABAP includes inside SLT. Because it uses ABAP code, you probably could create more complex transformations, but then you would interfere with the real-time nature and advantage of SLT.

So if you require proper data cleaning, you might rather want to consider using SAP Data Services.

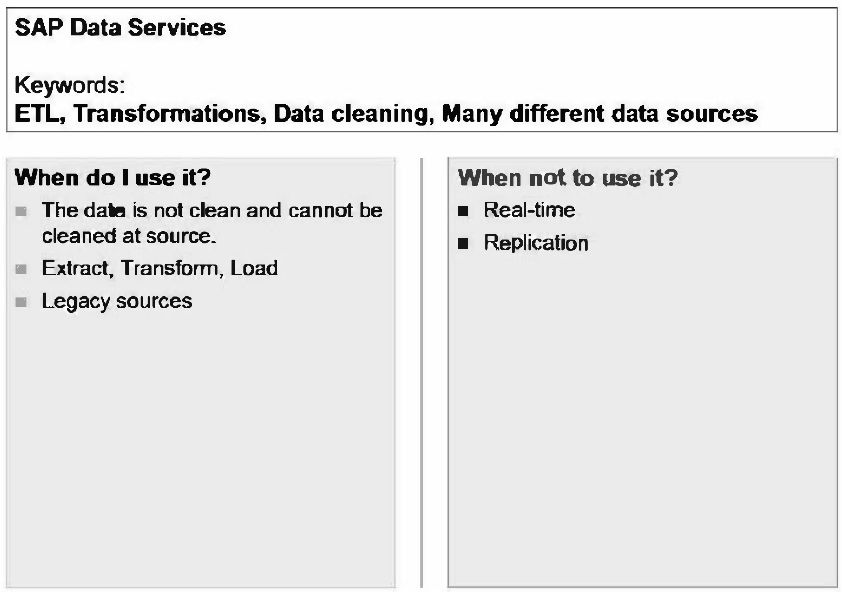


Figure 15: SAP Data Services

SAP Data Services

We know the old saying, "Garbage in, garbage out." If you have rubbish data going into your reporting system, you can expect rubbish data in your reports.

With SAP HANA this old saying changes a little. It is now, "Garbage in, garbage out fast!". :) Data Services is a good ETL tool. The "E" phase is extracting data from the source systems. Data Services can read many data sources, even the "obscure" one like old C●B●L copybooks! It can also read SAP systems, even via extractors. It cannot however use all the SAP extractors. If you need e.g. extractors that require activation afterwards, then DXC would be a better tool to use. During the "T" phase of ETL, Data Services can clean your data. There is no better SAP tool for doing this than Data Services. This eliminates the "garbage" portion of your reporting results. Then finally in the "L" phase of the ETL, Data Services will load the data into SAP HANA.

SDA – Smart Data Access

Keywords:
Virtual tables, Logical Data Warehouse (DW), Big Data, Reporting layer

When do I use it?

- Physical DW not feasible, Big Data
- Data from Hadoop
- Move back-end DB without affecting reports
- Small HANA system

When not to use it?

- Real-time
- Replication
- Data cleaning
- Full SAP HANA performance
- Pool or cluster tables

Figure 16: SDA – Smart Data Access

SDA – Smart Data Access

A few years ago people wanted to put all their data into a single data warehouse-type environment to analyze it there. This is sometimes called a "data lake", and is a physical data warehouse.

There are a few problems with this. You have to duplicate all your data, so now your data is double the size. You have to copy all that data via the network, which clogs up and slows your network. You need more storage. You basically doubled you cost and effort. The data in the new data warehouse is never up to date, and as such the data is never consistent. And the single data warehouse system itself cannot handle all the different types of data requirements, e.g. unstructured data, graph-engine data, key-value pairs, spatial data, huge variety of data, etc. These are some of the issues we deal with in the "Big Data" scenarios.

So people came up with the idea of a "logical data warehouse". You keep the data in your source systems, don't copy it via the network, don't store copies of it, etc. But you still create a single data warehouse, but in this case that "looks like" it contains all the data. This is where **Smart Data Access** comes in.

We get the data from the various source databases only when required. We get those bits of specific data, combine it with other data, and report on it. Another place to use it is when you plan to get rid of old legacy databases, or consolidate various databases. You wish to do this without affecting your end users and their reporting. In that case you use SAP HANA and SDA as a logical data warehouse. You connect to the various back-end databases, and end-users get their data via SAP HANA and SDA. You can now (carefully) start migrating or consolidating legacy databases with minimal disruption to your end users.

SRS – SAP Replication Server

Keywords:
Real-time, Log files, Replication, Disaster recovery replication

When do I use it?

- Real-time
- Replication
- Low impact on source systems
- Used for disaster recovery replication e.g. with SAP ASE

When not to use it?

- You require data cleaning
- Had problems with pool and cluster tables of older SAP systems

Figure 17: SAP Replication Server

SRS – SAP Replication Server

The big focus of SRS is on low-impact real-time replication. SRS works with both SAP and non-SAP source systems and support the many databases. These include SAP ASE, Microsoft SQL Server, IBM DB/2 and Oracle. The method that SRS uses for the real-time replication is log-based. It reads the log files of the source systems, and has very little impact on these source systems. So our previous example of the banks can use SRS instead SRS had a problem to read and use pool and cluster tables from older SAP systems.

The newer SAP systems are moving away from pool and cluster tables, e.g. SAP HANA and SAP ASE systems do not use them at all, so this becomes less of an issue with these newer systems.

SRS does not do filtering or transformation of data as you load the data into SAP HANA. If you require proper data cleaning, you should consider perhaps using SAP Data Services.

SDI – Smart Data Integration

Keywords: **Same as SAP Replication Server, but built-in and limited to SAP HANA only**

When do I use it?

- Same as SAP Replication Server
- Limited to SAP HANA only (will also bring price down)

When not to use it?

- Same as SAP Replication Server
- Need SAP Replication Server for non-HANA scenarios as well

Figure 18: SDI

SDI – Smart Data Integration

The same advantages and disadvantages of SRS (SAP Replication Server) applies to SDI (Smart Data Integration). This is because SDI is similar to SRS implemented inside SAP HANA when using the log file adapters. You buy this as an additional feature for SAP HANA, and as such it has separate pricing.

The pricing will be better than a full SRS server as SDI is limited to working with SAP HANA only, whereas SRS will be a full server working with multiple scenarios. If you require SRS functionality, but only in an SAP HANA context, it will work you out cheaper to rather use SDI. If however you are not limited to SAP HANA only, you should rather use SRS separately.

SDI can have additional adapters (e.g. to Twitter) in addition to the log file adapters.

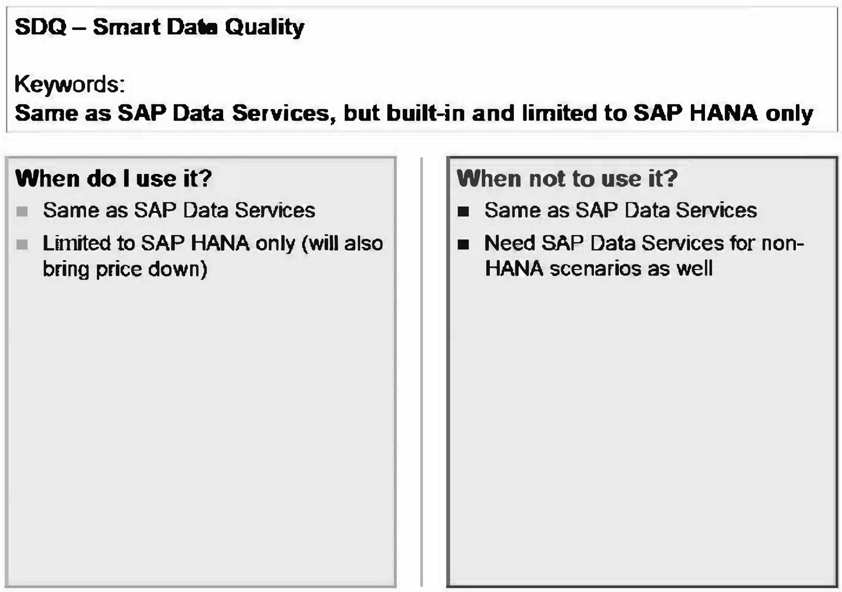


Figure 19: SDQ

SDQ – Smart Data Quality

The same advantages and disadvantages of SAP Data Services applies to SDQ (Smart Data Quality). This is because SDQ is basically Data Services functionality implemented inside SAP HANA. You buy this as an additional feature for SAP HANA, and as such it has separate pricing.

The pricing will be better than a full Data Services server as SDQ is limited to working with SAP HANA only, whereas Data Services will be a full server working with multiple scenarios. If you require Data Services functionality, but only in an SAP HANA context, it will work you out cheaper to rather use SDQ. If however you are not limited to SAP HANA only, you should rather use Data Services separately. In some sense the traditional ETL process has to be renamed to ELT when using SDQ, as it will extract the data from source and then load and transform it inside SAP HANA.

Note: Enterprise Information Management (EIM) it is a umbrella term for SDI and SDQ

SDS – Smart Data Streaming

Keywords: **Streaming data, real-time, events**

When do I use it?

- Analyse real-time streaming data for events, e.g. in manufacturing or production

When not to use it?

- Replication
- Data cleaning

Figure 20: Smart Data Streaming

SDS – Smart Data Streaming

SDS is the same as the old Sybase ESP (Event Stream Processor), but just built into SAP HANA now.

The same advantages and disadvantages of ESP applies to SDS (Smart Data Streaming). This is because SDS is basically ESP implemented inside SAP HANA. You buy this as an additional feature for SAP HANA, and as such it has separate pricing. The pricing should be better than a full ESP server as SDS is limited to working with SAP HANA only, whereas ESP will be a full server working with multiple scenarios. If you require ESP functionality, but only in an SAP HANA context, it should work you out cheaper to rather use SDS. If however you are not limited to SAP HANA only, you should rather use ESP separately.

The use case for SDS (or ESP) is very specific. If you have streaming data e.g. financial data from stock exchanges, or operational data like temperatures and pressure and flows from manufacturing plants or oil refineries, you can only use SDS. As such it is easier to decide when to use SDS!

DXC

Keywords:
SAP Extractors, SAP Business Warehouse

When do I use it?	When not to use it?
You need to use SAP extractors (and Data Services cannot read the ones you need)	Real-time Replication Data cleaning Non-SAP source systems

Figure 21: DXC

DXC – Direct Extractor Connection

The big drive for DXC is when you want to use complex extractors from SAP systems to load data into SAP HANA. Especially when the extractors require activation, Data Services might not be able to use these extractors. The main business case for this is when you would like to get some data from an SAP Business Suite system e.g. financial data.

You can replicate the multiple financial tables from the SAP source systems using SLT, but you will end up with 10+ tables inside SAP HANA. You then have to create **a complex data model** to join all the tables, and manipulate the data. Using the extractors via DXC, you get the financial data delivered to you in SAP HANA in a "nice packaged" way that is immediately usable, where you don't have to create a complex data model first. The disadvantage is that these extractors are not available in real-time.

Many times you will want to use extractors that are already used to extract data to an existing BW system. Most extractors can however create multiple delta queues, so you can re-use the same extractors with a new delta queues to provide data to SAP HANA. An exception to this is e.g. Banking Services extractors that can only have a single delta queue.

Flat files or Excel sheets

Keywords: **Proof of concept projects, Testing, Quick, Simple**

When do I use it?

- One-time loading of data into test systems or proof-of-concept projects
- Simple, Quick
- Source only produces flat files

When not to use it?

- Real-time
- Replication
- Data cleaning

Figure 22: Flatfile loading

Flat files or Microsoft Excel datasheets

Flat files or Excel datasheets are the simplest way to provision data to SAP HANA.

Especially in proof-of-concept (POC) or sandbox testing projects, you can waste lots of time to connect SAP HANA to the various systems. By the time you have put a server into the data centre and connected it to the source system, the time allocated for your POC will be finished already. As the point of POC project is normally not to proof that SAP HANA can integrate to the other systems via the provisioning tools we're discussing, but rather that SAP HANA can deliver some functionality we could not do previously.

As such it makes more sense to ask the business to provide you with the data in flat files, and you just load the data into SAP HANA for the POC or sandbox development. Sometimes you have system that have such exotic interfaces that creating flat files for interfacing is the best (or even only) way.

Just watch out for security concerns. Many companies have a rule that "data at rest" (i.e. on disk) should be encrypted. In this case the security policies will prevent you from using this method.

Web Services

Keywords:
REST, Odata, Web Services, XS development

When do I use it?

- Source systems expose their data via web services

When not to use it?

- You do not want to do any development
- Your SAP HANA license does not allow this

Figure 23: WEB services

Web Services

We don't really discuss this option in this course, but you can of course consume data from source system that provide it via web services. You can consume these web services using the XS development environment inside SAP HANA. You have to ensure though that you have the proper SAP HANA license in place for such development.

Which provisioning tool do I use?

In very broad terms:

- Real-time, Replication: SLT, SRS or SDI
- Pool or cluster tables: SLT
- ETL, Data cleaning: Data Services or SDQ
- Extractors: Data Services or DXC
- Streaming data: SAP ESP or SDS
- Federation, Big Data: SDA
- POC, Development, Testing: Flat files

Figure 24: Summary

Which tool do I use? There are of course the exceptions, but the slide gives you a good overview of when to use which tool.

Lesson Summary

You should now be able to:

- Explain the acronym ETL
- State which methods of data provisioning can be used for SAP HANA

Unit Summary

You should now be able to:

- Understand the basic concepts and capabilities of SAP HANA
- Explain the acronym ETL
- State which methods of data provisioning can be used for SAP HANA

Unit 2

SAP LT - SAP Landscape Transformation

Unit Overview

In this unit we cover the key areas of SAP LT including deployment options, configuration, replication and transformation.

Unit Objectives

After completing this unit, you will be able to:

- Explain the positioning of SAP LT Replication Server
- Describe the key concepts and features
- Describe the key steps to create a configuration
- Replicate data to SAP HANA using SLT
- Explain how to change advanced replication settings for certain configurations of SAP LT Replication Server
- Explain the basics of the SLT based transformation concepts
- Know which types of transformation rules exist
- Describe the configuration and monitoring dashboard
- Explain the concept of mass transfer IDs
- Leverage the SLT Cockpit

Unit Contents

Lesson: Positioning and key Concepts

Lesson Overview

This lesson is an introduction to SAP LT, its positioning and key benefits

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the positioning of SAP LT Replication Server
- Describe the key concepts and features

Business Example

In order to explain to your client the benefits of SAP LT Replication Server you want to detail the overview key concepts and features so that you can make an informed decision if SLT is the correct path for your client to follow in order to load data into HANA. The SAP Landscape Transformation (LT) Replication Server is the SAP technology that allows you to load and replicate data in real-time from SAP source systems and non-SAP source systems to an SAP HANA environment

SAP Landscape Transformation

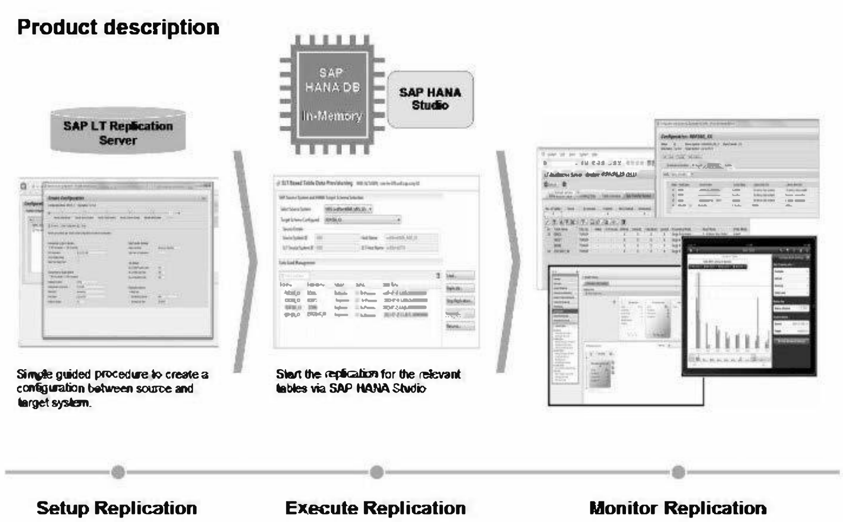


Figure 25: Product description

SAP Landscape Transformation Replication Server (aka 'SLT')

is standard software to move data in real-time between different systems within the same network, wide area networks, or into the cloud to have the information at the right place at the right point of time.

SLT helps to feed analytical systems with up-to-date business information from the productive system landscape, support the acceleration of large volume transactions executed in SAP HANA, enables real-time reporting and minimizes transfer volume for SAP BW and enables the synchronization between different systems.

Replication

Real-time or scheduled

Delta capturing to minimize transfer volume

Transformation

Filtering by using selective criteria

Adjustment of tables

Conversion of data, e.g. to make sensitive data anonymous

Installation & operation

Deeply integrated within SAP landscapes to reuse existing installation and monitoring capabilities

Monitoring with SAP Solution Manager 7.1 SP5 onwards

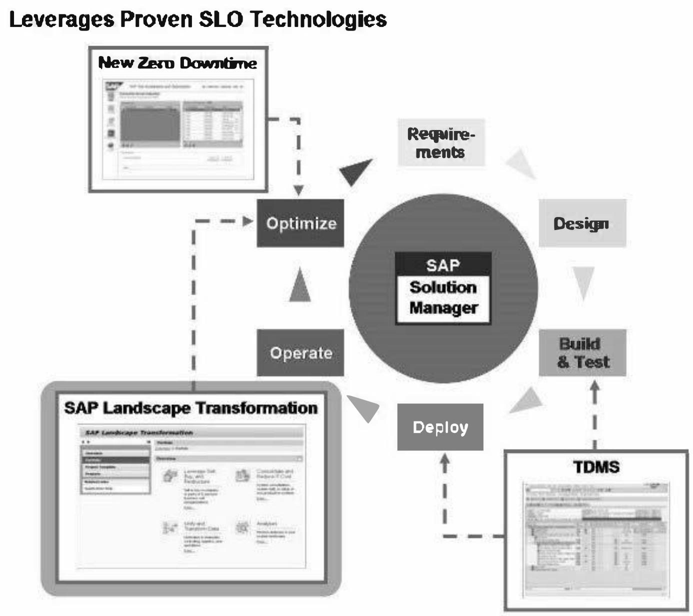


Figure 26: Leverages Proven SLO Technologies

SLT is built from well established and proven SAP technology for many years with large numbers of customers already using this technology today.

SAP Landscapc ●ptimizations (SL●) is a family of tools to support a varicty of requirements as seen in the diagram.

This was the used as the basis for what we have today and has been enhanced to provide support for SAP HANA data replication scenarios with HANA Studio support for controlling replication from SAP HANA.

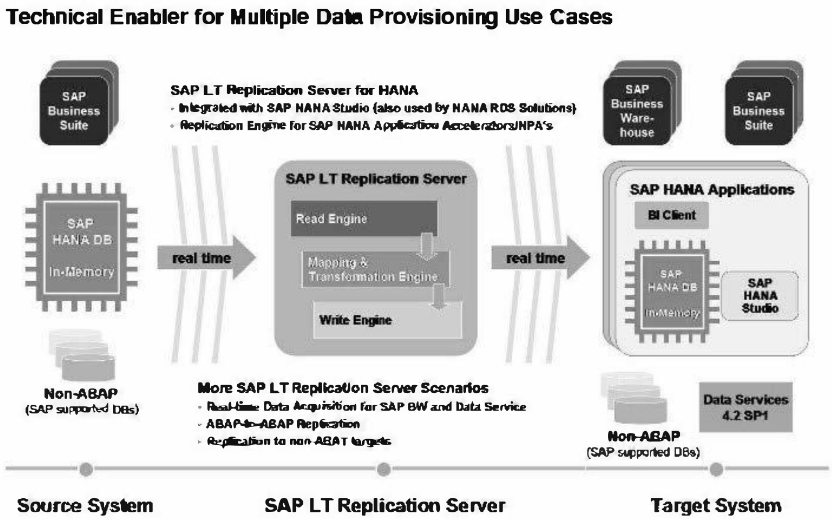


Figure 27: Technical Enabler for Multiple Data Provisioning Use Cases

SAP LT Replication Server is used to load and replicate data in scheduled or real-time mode from an ABAP or non-ABAP based source system into SAP HANA, SAP Data Services, SAP BW or any other ABAP based system.

Installed Base	4407 customer landscapes running with SAP LT Replication Server
Highest Initial Load	26 billion records table (89 hours)
Most Replicated Tables	12.000 tables in one landscape, 500 tables in one configuration
Most connected SAP Systems	70 connected SAP systems on one SAP LT Replication Server
Most connected non-SAP Systems	45 connected non-SAP systems on one SAP LT Replication Server
Highest Change Rate	> 20 Mio records per hour per table

Figure 28: SAP Landscape Transformation Replication Server

SLT can be scaled to handle very large transaction volumes. See the table for some examples. Bear in mind these figures will only improve over time and were correct as of 2015 Q1.

SLT Benefits

- **Natively leverages SAP ABAP Data Objects for replication on application level:** Data load, as well as real time or scheduled replication of:
  - Pool, Cluster, INDX*), HR*) tables
  - Unicode & non-Unicode
  - Replication of views
  - Supports all SAP Systems, releases starting from SAP R/3 4.5c
  - Table-based and application-based replication
- **High Volume change data capturing with almost zero impact and** point-in-time recovery capability
- **Complex transformation capabilities**, such as filtering data structure adjustments, data conversion, anonymisation of sensitive data, etc.
- **Deeply integrated within SAP landscapes to reuse existing administration and monitoring capabilities**, i.e of SAP Solution Manager, SAP NW Application Server and SAP HANA Studio
- Connects to **SAP HANA data sources with optimized change capturing**
- Connects to **native Databases as data sources** (according to the SAP NW Product Availability Matrix)
- Connects to **native Databases as data targests** (according to the SAP NW Product Availability Matrix) via project engagement
- **Real-time or scheduled data replication into SAP BW reduces the amount of overnight data uploads. Delta updates on BW DataSources without delta mechanism**, for ABAP-based systems as well as non-ABAP based systems on all SAP supported DB versions (according to PAM).

Figure 29: SLT Benefits

What are the key benefits ?

- Allows real-time (and scheduled) data replication Ability to migrate data into HANA format while replicating data in real-time
- „Unlimited“ release coverage (from SAP R/3 4.6C onwards) sourcing data from ABAP based SAP applications
- Handling of cluster and pool tables
- Automatically non-Unicode to Unicode conversion during load/replication
- Table settings and transformation capabilities (e.g. data filtering, enrich table structure, anonymize data, etc.)
- Fully integrated with SAP HANA Studio
- Enhanced monitoring capabilities via SAP Solution Manager 7.1 SP5 & mobile app SAP Replication Manager

The SAP LT Replication Server uses a trigger-based replication approach to pass data from the source system to the target system.

Note: For non-SAP source systems, the customer database license needs to cover a permanent database connection with 3rd party products like SAP LT Replication Server.

System	Purpose
SAP Source System	The source system tracks database changes by using database triggers. It records information about changes in the logging tables. Read modules (located on the SAP source system) transfer the data from the source system to the SAP LT Replication Server. The relevant data is read from the application tables.
Non-SAP Source System	The non-SAP source system tracks database changes by using database triggers. It records information about changes in the logging tables. Read modules (located at the SAP LT Replication Server) transfer the data from the non-SAP source system to the SAP LT Replication Server. The relevant data is read from the application tables.
SAP Landscape Transformation Replication Server	An SAP system that facilitates the replication of data from one or more source systems to one or more target systems. The source systems can be SAP or non-SAP systems.
SAP HANA System	The SAP HANA system contains the SAP HANA database. It is used to store the replicated data. The SAP LT Replication Server and the SAP HANA system communicate by means of a database connection.

The SAP LT Replication Server can be used for replication from SAP sources and non-SAP sources to the HANA system. For SAP sources, the SAP LT Replication Server can either be installed within the source system or in a separate SAP system.

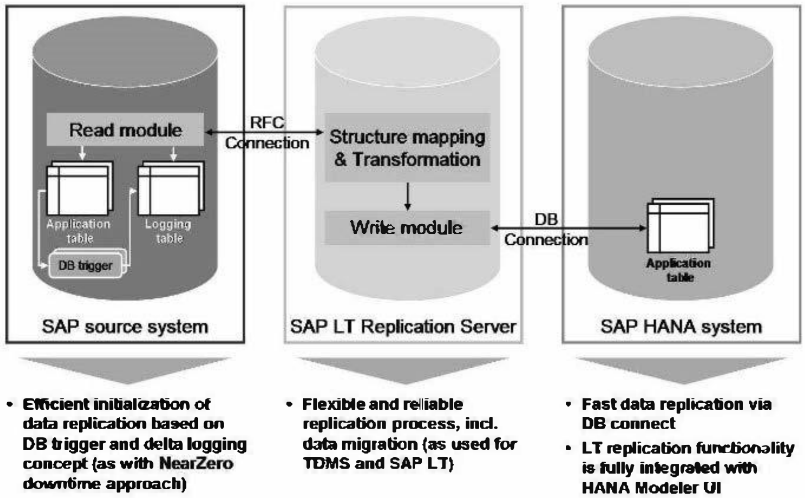
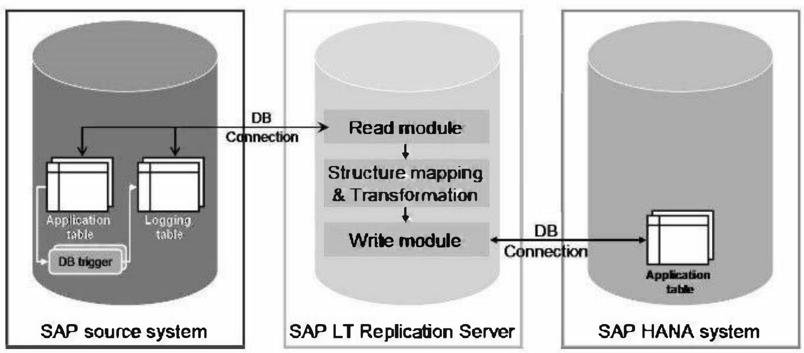


Figure 30: Architecture and Key Building Blocks - Positioning and Key Concepts



In a first step, SAP LT Replication Server transfers all metadata table definitions from the non-SAP source system to the HANA system. From the HANA Studio perspective, non-SAP source replication works as for SAP sources. When a table replication is started, SAP LT Replication Server creates logging tables within the source system. As a difference, the read modules are created in the SAP LT Replication Server. The connection the non-SAP source system is established as a database connection.

Figure 31: Architecture for Non-SAP Source Replication - Positioning and Key Concepts

In a first step, SAP LT Replication Server transfers all metadata table definitions from the non-SAP source system to the HANA system.

Note: From the HANA Studio perspective, non-SAP source replication works as for SAP sources

When a table replication is started from a non-SAP source system, SAP LT Replication Server creates logging tables within the source system. As a difference, the read modules are created in the SAP LT Replication Server. The connection to the non-SAP source system is established as a database connection.

The relevant information to create the connection between the source system, the SAP LT Replication Server, and the SAP HANA system is specified within the SAP LT Replication Server as a Configuration.

In the Configuration & Monitoring Dashboard (transaction LTR), you can define a new configuration. You use a configuration to load and replicate data from one source system to one target database schema of a HANA system (1:1), or from multiple source systems to one target database schema of an SAP HANA system (N:1). Furthermore, it is possible to load and replicate data from one source system to multiple (up to 4) target database schemas of one or more HANA systems (1:N).

You can also specify the type of data load and replication - either in real-time, or scheduled by time or by interval.

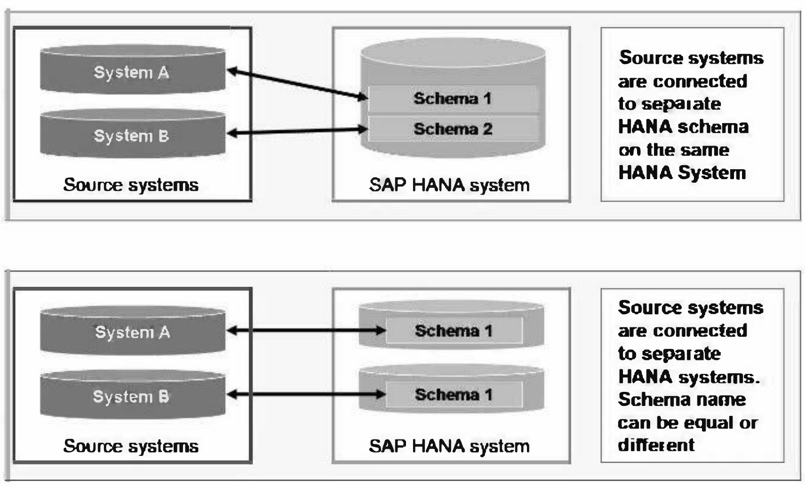


Figure 32: Multi System Support 1/3 - Positioning and Key Concepts

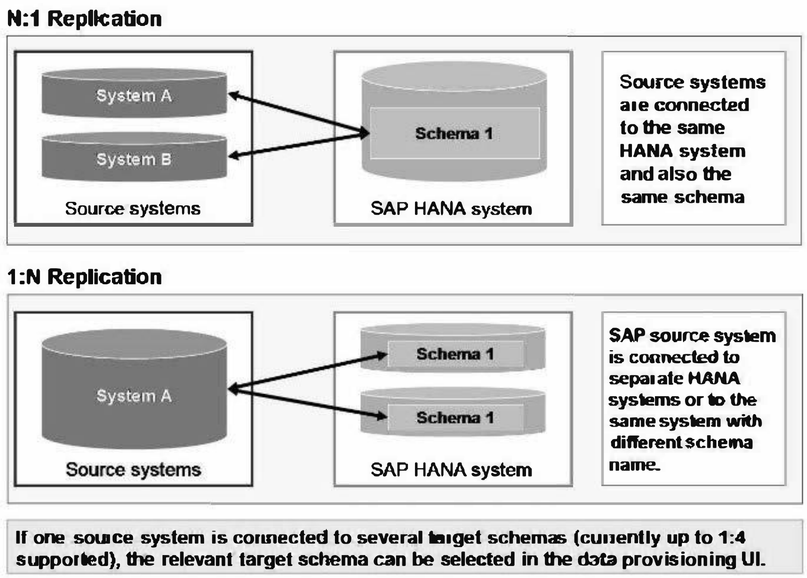


Figure 33: Multi-System Support 2/3 - Positioning and Key Concepts

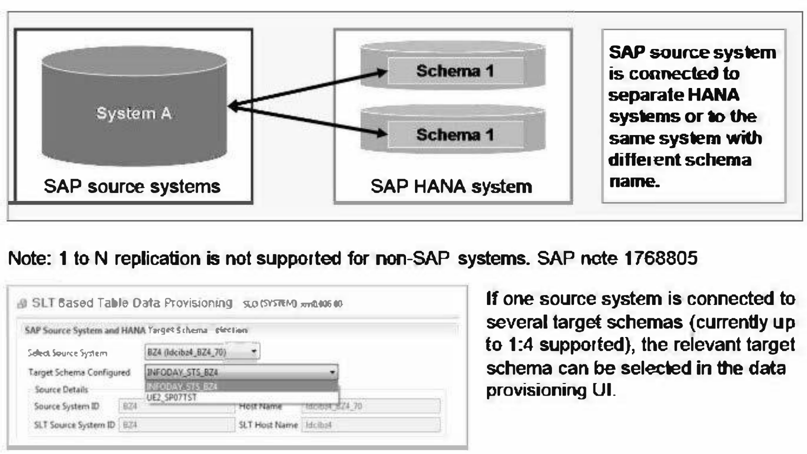


Figure 34: Multi-System Support 3/3 - handling 1:N replication

Lesson Summary

You should now be able to:

- Explain the positioning of SAP LT Replication Server
- Describe the key concepts and features

Lesson: Configuration Overview

Lesson Overview

This lesson is an introduction to SAP LT, its positioning and key benefits

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the key steps to create a configuration

Business Example

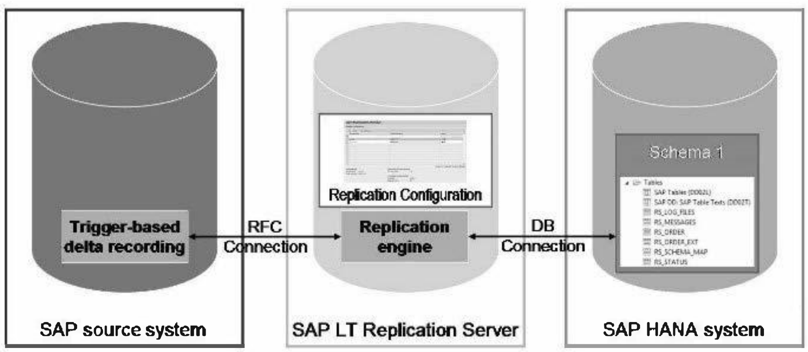
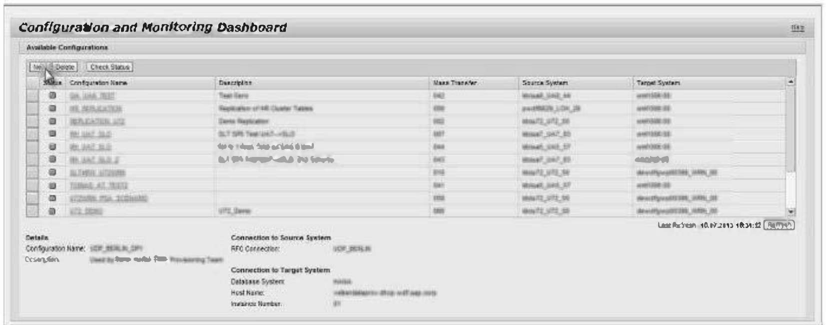


Figure 35: Define Configuration / Schema

A new configuration can be created in the LT Configuration and Monitoring Dashboard. In that step, the connection between the source and the HANA system is established and the target schema will be created (if it does not exist already). In addition, replication control tables are created and table lists are replicated from the source system. Also, the required roles and GRANT/REV●KE procedures are generated.

Call transaction LTR to start the Web interface for the creation of replication configurations. Click the *new* button to create a new configuration.



A replication configuration can be created via web interface, via transaction LTR.

Figure 36: Transaction LTR

Once you create a new configuration, a wizard will be launched to guide you through the setup of the configuration.

Specify General Data

In the first step, you need to define a configuration name, which will be also the name of the schema created in the HANA database. You can then enter a free text description.

Specify Source System

In the second step, you must specify details for the source system. In case of an SAP source system, you have to choose *RFC Connection* and specify the name of the RFC connection pointing to the client you want to replicate. In case you mark the flag *Read from single client*, then only the client specified in the RFC connection will be replicated. If you mark the *Allow Multiple Usage* flag, then you will allow the usage of the source system in different configurations (1:N replication).

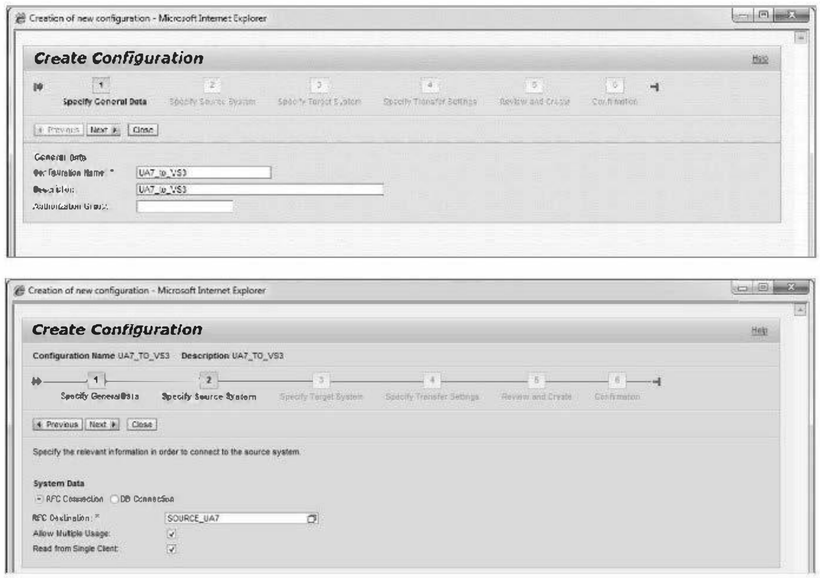
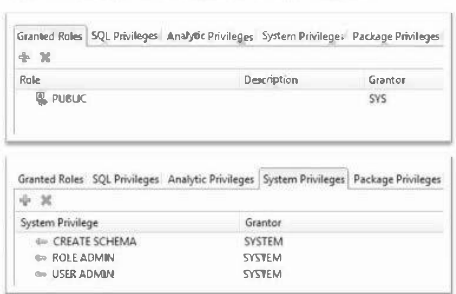


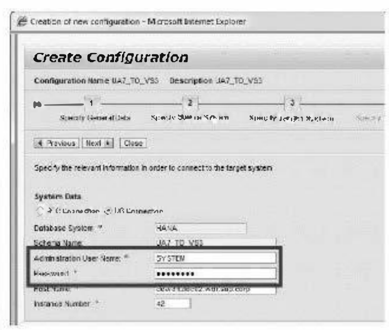
Figure 37: Configuration Wizard 1/3

Specify Target System

To connect to the HANA system and to create the relevant objects, an user with the required privileges needs to be defined in the connection details for the HANA system.

You should create a HANA user with the required system privileges, or use the SYSTEM user:





See also SAP Note 1635735

Figure 38: User for Connection to the SAP HANA System

In the third step of the wizard, you can specify the connection details for the target system. In order to replicate the data into a HANA database, you choose *DB Connection* and specify the *Host name* and *Instance number* of the HANA server. A user and password also need to be specified; this is the *initial user* whose authorization requirements are described in the SLT Security Guide.

Specify Transfer Settings

Here, you can specify the number of jobs to be used during replication, during the initial load, and during the calculation of the access plans. You can further define the type of replication by choosing among

- Real Time
- Schedule by Interval
- Schedule by Time

Optionally, you can define the data class for the creation of logging tables. If no data class is defined, the logging tables will be created in the same table space as the original table. An own table space is recommended for easier monitoring of the table sizes of the logging tables.

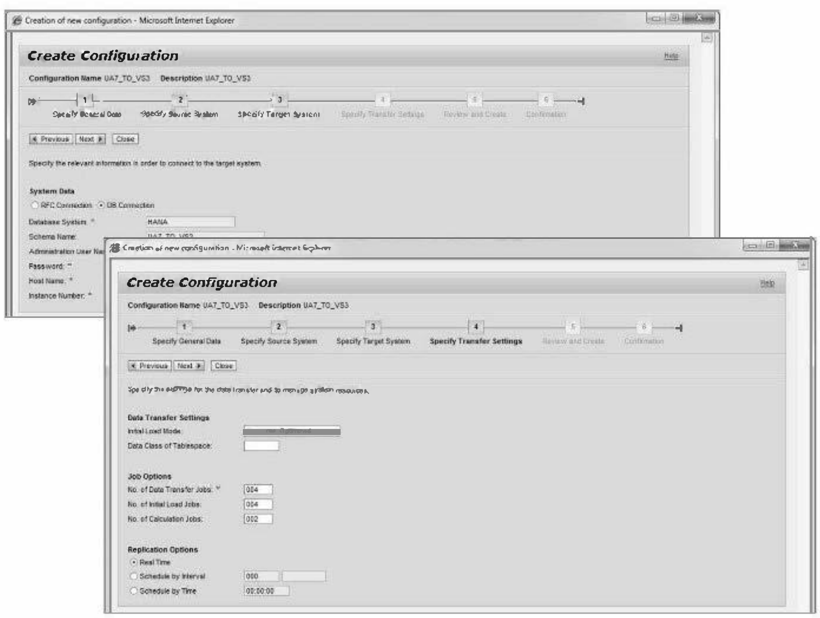


Figure 39: Configuration Wizard 2/3

Review and Create

You can finally review the entered settings and confirm the creation of a new replication configuration.

Confirmation

In the last step, you get a confirmation for the creation of the new configuration and you can then close the creation wizard.

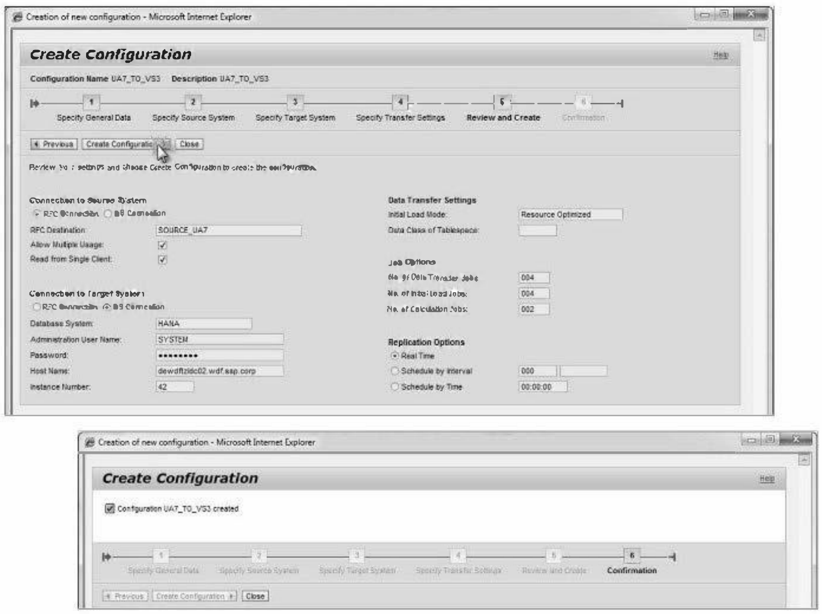


Figure 40: Configuration Wizard 3/3

Alternatively, the creation of a replication scenario can be done completely from the ABAP interface, within the replication cockpit, transaction LTRC.

The creation of a configuration can be done within transaction LTRC.

For SLT DMIS 2011 SP8 we now have an improved interface for configuration creation in the replication cockpit (LTRC). A wizard similar to the web dynpro method is used.

Figure 41: The Tree Structure in HANA Studio

When the creation of a *new configuration* is confirmed, the following actions are performed automatically:

- Configuration settings are saved on the LT Replication Server.
- New user and schema are created on the HANA system with the defined target schema name (not performed if an existing schema is reused).
- Replication control tables (RS_& tables) are created in the target schema.
- User roles for the target schema are created:
  - <target_schema>_DATA_PROV → Role to manage data provisioning
  - <target_schema>_POWER_USER → Contains all SQL privileges of the target schema
  - <target_schema>_USER_ADMIN → Role to execute authority procedures (see below)
- A procedure to grant (RS_GRANT_ACCESS) or revoke (RS_REVOKE_ACCESS) are created in the target schema.
- Replication of tables DD02L (stores the table list), DD02T (stores the table short descriptions), and DD08L (R/3 DD: relationship definitions) is started automatically. Once those tables are replicated, the HANA studio knows which tables are available in the source system.
- SYS_REPL and table RS_REPLICATION_COMPONENTS are created (if they do not exist already based on a previous configuration).
- Replication is registered in table RS_REPLICATION_COMPONENTS.

Most relevant replication control tables are:

- **RS_ORDER**
  Table which temporarily stores the data provisioning requests (e.g. replication) to be processed by the LT Replication.
- **RS_STATUS**
  Stores the data provisioning status (current and historical) for each table processed by the LT Replication.

For each target schema, three roles are generated which can assigned to a user:

Name	Role	Purpose
Data Provisioning	<SCHEMA>_DATA_PROV	Data provisioning - deploy replication orders
Power User	<SCHEMA>_POWER_USER	Power user - allow modification within the replication schema
User Admin	<SCHEMA>_USER_ADMIN	Can call the RS_GRANT_* and RS_REVOKE_* procedures

To assign or revoke individual authorities, there are also two respective procedures generated for each target schema. To execute these procedures the user admin role must be assigned.

Granting Access Rights

```
SET SCHEMA <REPLICATION SCHEMA>;
CALL RS_GRANT_ACCESS
        ('TABLE NAME',
        'USER OR ROLE NAME',
        'X', -- to grant SELECT
        'X', -- to grant INSERT
        'X', -- to grant UPDATE
        'X') -- to grant DELETE
```

Revoking Access Rights

```
SET SCHEMA <REPLICATION SCHEMA>;
CALL RS_REVOKE_ACCESS
        ('TABLE NAME',
        'USER OR ROLE NAME',
        'X', -- to revoke SELECT
        'X', -- to revoke INSERT
        'X', -- to revoke UPDATE
        'X') -- to revoke DELETE
```

Figure 42: Use of Advanced Role and Security Concept

The **REPLICATION SCHEMA** is the schema where the table is located that is subject to altering access rights.

TABLE NAME is the name of the table in the schema. **USER OR ROLE NAME** is the name of the user or role that shall be granted or revoked access. Use 'X' to grant or revoke a specific right, use ' ' to leave the rights unchanged.

Lesson Summary

You should now be able to:

- Describe the key steps to create a configuration

Lesson: Data Replication at a Glance

Lesson Overview

This lesson covers the following topics:

- Load and replicate data
- Suspend and resume data replication of certain tables
- Stop and Restart the master job of a configuration in SLT

Lesson Objectives

After completing this lesson, you will be able to:

- Replicate data to SAP HANA using SLT

Business Example

SAP LT Replication is set up at your client site. You now need to load some data and make sure you can suspend and resume the replication as appropriate.

All further data provisioning steps are executed from the HANA Studio. Therefore, switch to the HANA Studio, choose the perspective *Information Modeler* and start the quick launch.

Select your system and start the *Data Provisioning* via the link in section *DATA*.

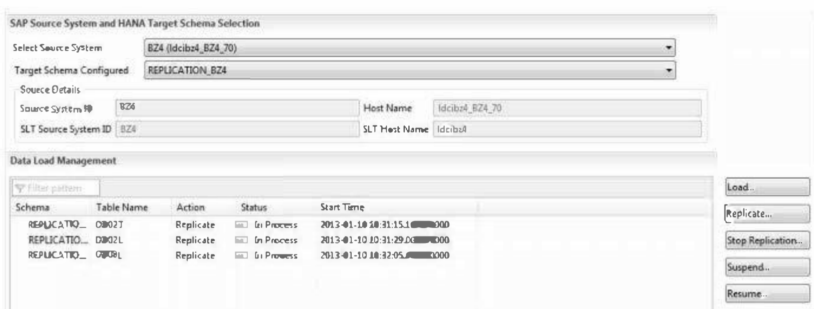


Figure 43: Launch Data Provisioning UI

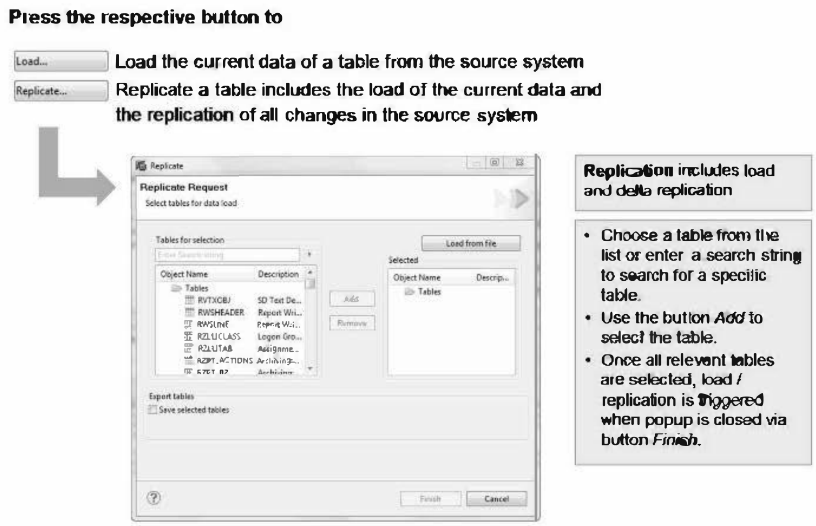


Figure 44: Start Load / Replication

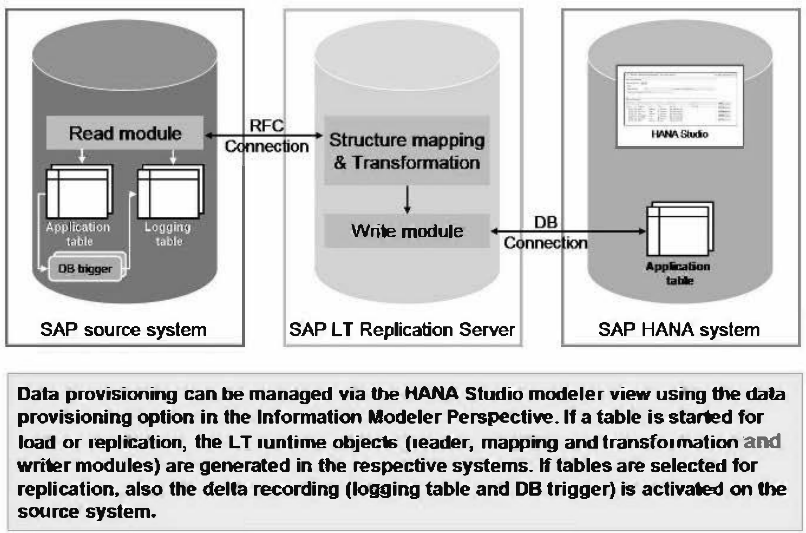


Figure 45: Stop / Suspend Replication, Generated LT Runtime

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Press the respective button to

Stop Replication...	Stop the replication and also to stop delta recording for that table (deletes DB trigger!)
Suspend...	Suspend data replication but keep delta recording active
Resume...	Resume a previously suspended data replication

The table selection popup will look similar as for *Load* and *Replication*. In case of *Stop Replication* or *Suspend*, only those tables can be selected which are already in replication mode. In case of *Resume* only those tables can be selected which are in suspend mode.

Please be aware that in case a replication is stopped and started again, the corresponding table will be dropped. The initial load must be repeated as delta recording was deactivated for a certain time and changes might not be recorded. So in case you only want to pause the delta replication, use the mode Suspend and Resume as delta recording is not deactivated and the replication can be continued without a need of a new initial data load.

Figure 46: Stop / Suspend Replication

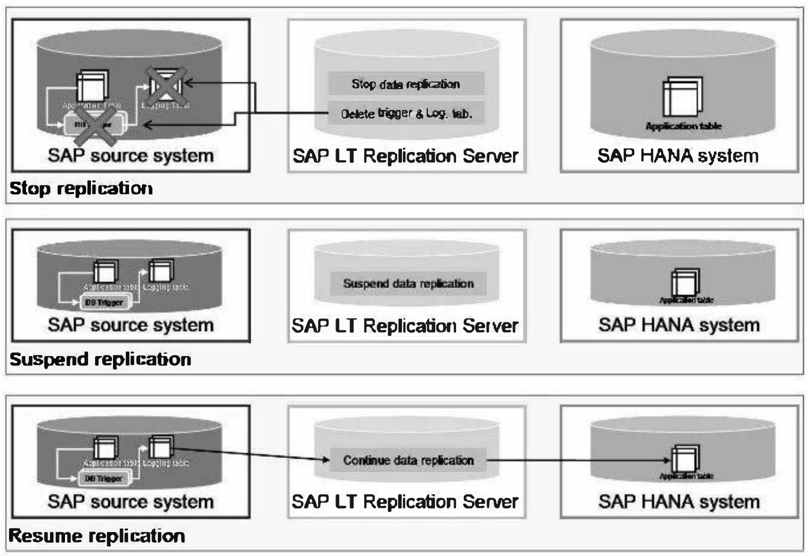


Figure 47: Stop/Suspend Replication – Executed Activities

Alternatively the Data Provisioning activities can be triggered within the replication cockpit, transaction LTRC as in the following figure.

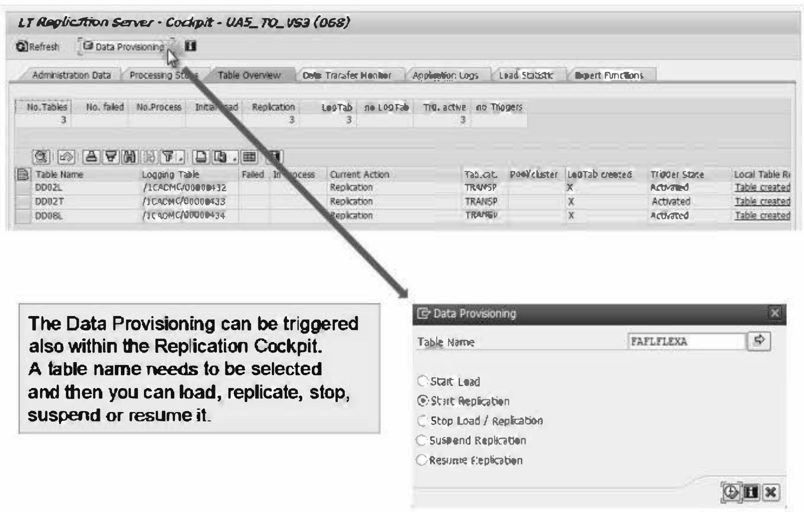


Figure 48: Data Provisioning from the Replication Cockpit

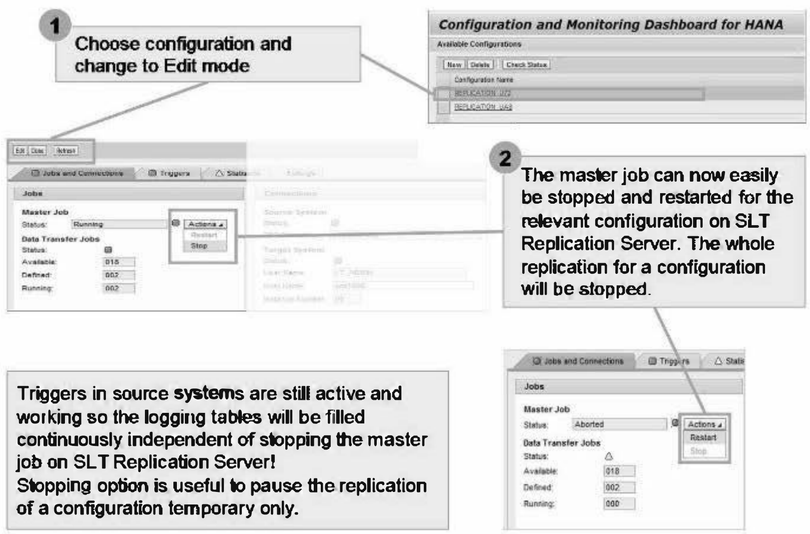


Figure 49: Configuration and Monitoring Dashboard Master Job Settings – Stop and Restart

Exercise 1: Start a replication using SAP HANA Studio

Exercise Objectives

After completing this exercise, you will be able to:

- Start and stop a replication using the SAP HANA Studio

Business Example

You need to ensure your SAP HANA data is kept in sync with your ERP data

Task:

Replication of a table

Hint: ●ne note regarding the numbers inside the graphics. Sometimes there could be a gap between two consecutively numbers. Please do not pay attention it . The only important thing is the connection between number and its description.

Field Name	Value
Host Name:	wdflbmt7215.wdf.sap.corp
Instance Number:	00
Description:	HDB
HDB User Name:	SYSTEM
HDB Password:	W3lcome123
Source tables search field	ZSFLIGHT_## (replace ## with your two digit group number)
Transaction	SE38
Program	Z_SFLIGHT
User Number (P_GROUP)	## (replace ## with your two digit group number

Figure 50: The system informations

1. Launch the SAP HANA Studio and setup a connection to the SAP HANA host.

Continued on next page

2. Replicate the table SFLIGHT_## from the SAP ERP source system to SAP HANA using the SAP HANA studio

3.

4.

Solution 1: Start a replication using SAP HANA Studio

Task:

Replication of a table

Hint: One note regarding the numbers inside the graphics. Sometimes there could be a gap between two consecutively numbers. Please do not pay attention it . The only important thing is the connection between number and its description.

Field Name	Value
Host Name:	wdflbmt7215.wdf.sap.corp
Instance Number:	00
Description:	HDB
HDB User Name:	SYSTEM
HDB Password:	W3lcome123
Source tables search field	ZSFLIGHT_## (replace ## with your two digit group number)
Transaction	SE38
Program	Z_SFLIGHT
User Number (P_GROUP)	## (replace ## with your two digit group number

Figure 51: The system informations

1. Launch the SAP HANA Studio and setup a connection to the SAP HANA host.

   a) When you launch SAP HANA Studio for the first time after installation , the **Secure Storage** is created with a generated master password !

Continued on next page

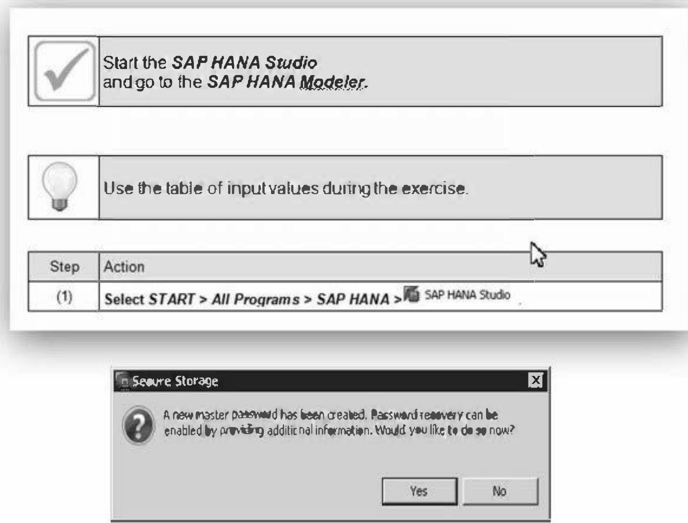


Figure 52: Step 2

b) When you launch SAP HANA Studio on your desktop for the first time, the Welcome page will be displayed showing the overview of the major SAP HANA Studio perspectives .

c) Click the Preferences menu item

Continued on next page

Step	Action
(2)	

Step	Action
(3)	Click the Window menu item to execute it.

Figure 53: Steps 2 and 3

Step	Action
(4)	Click the Preferences menu item to execute it.

Figure 54: Step 4

Continued on next page

Step	Action
(5)	Open the folder General by clicking the Open folder icon.

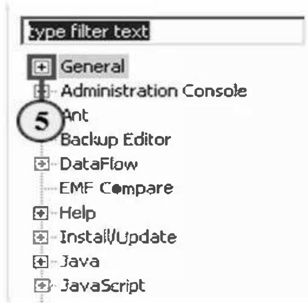


Figure 55: Step 5

Step	Action
(6)	Click Network Connections .

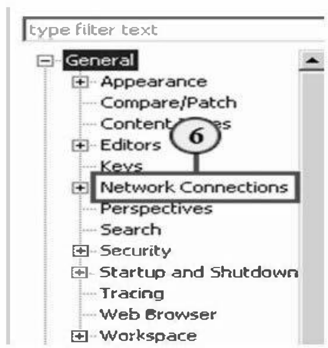


Figure 56: Step 6

Continued on next page

Step	Action
(7)	Click Active Provider: .

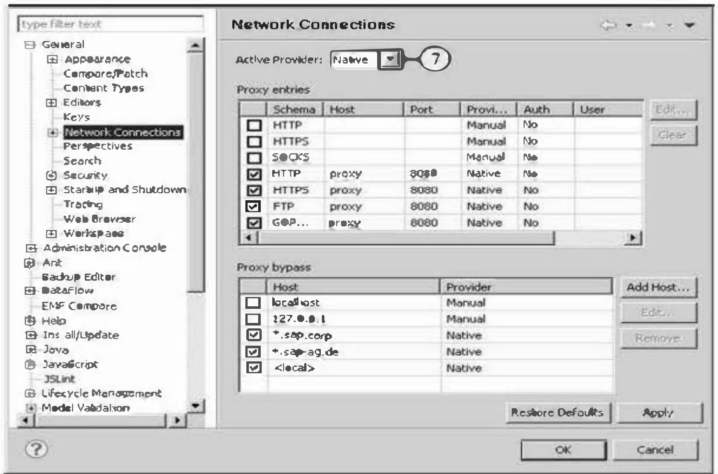
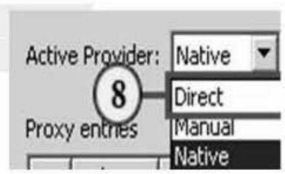


Figure 57: Step 7

Step	Action
(8)	Select the entry Direct by clicking it.



Step	Action
(9)	Click Apply .
(10)	Click OK .

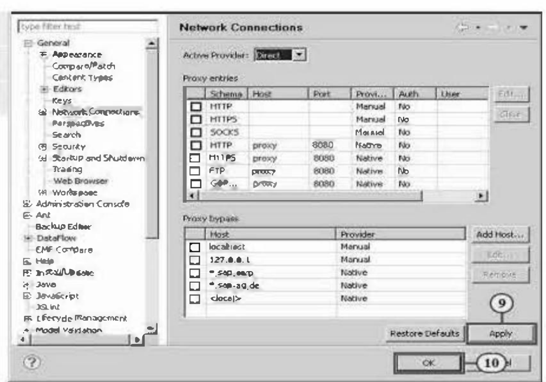


Figure 58: Step 8, 9 and 10

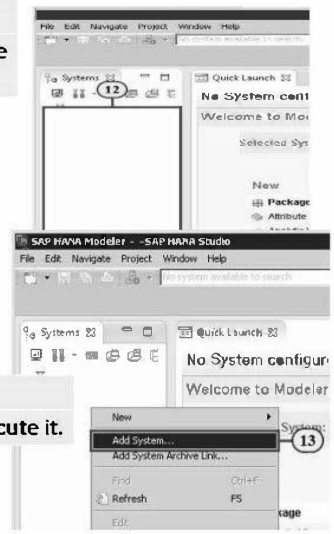
Continued on next page

Step	Action
(11)	Click .

Add a new system connection in SAP HANA Studio and logon with the SLT_USER to the SAP HANA system.

Figure 59: Step 11

Step	Action
(12)	Click on Systems Navigation Pane with the right mouse button.



Step	Action
(13)	Click the Add System... menu item to execute it.

Figure 60: Step 12 and 13

Continued on next page

Step	Action
(14)	Enter **wdflbmt7215.wdf.sap.corp** in the Host Name: box. Confirm your entry by pressing the Tab key.
(15)	Enter **00** in the Instance Number: box.
(16)	Enter **HDB in the Description:** box.
(17)	Click Next.

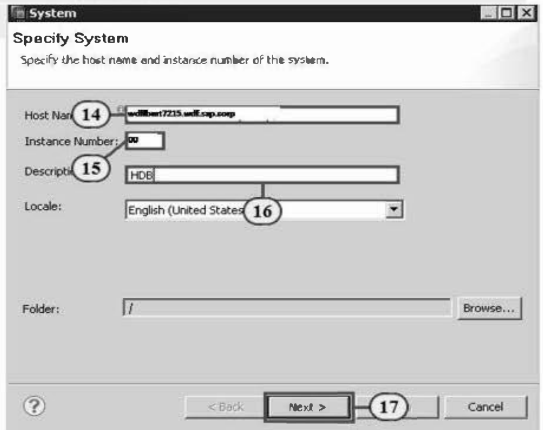


Figure 61: Step 14 - 17

Step	Action
(18)	Enter SYSTEM in the HDB User Name: box.
(19)	Enter **W3lcome123 in the HDB Password:** box.

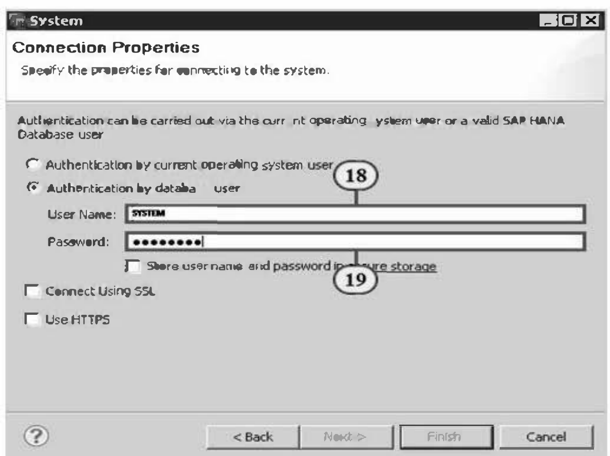


Figure 62: Step 18 - 19

Continued on next page

Step	Action
(20)	Click Store user name and password in secure storage .
(21)	Click Finish.

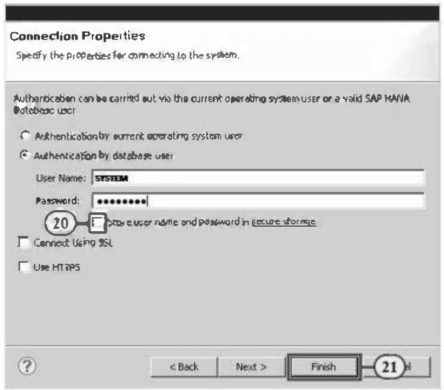


Figure 63: Step 20 and 21

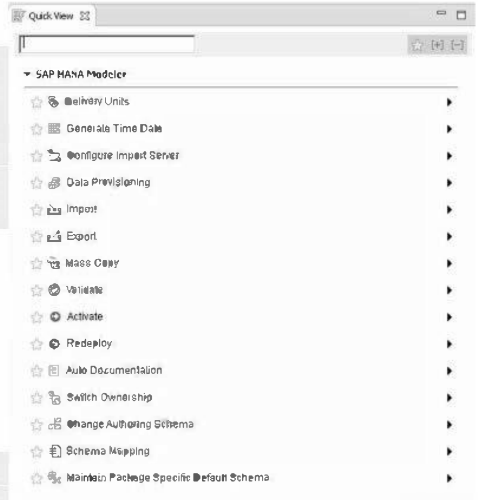
2. Replicate the table SFLIGHT_## from the SAP ERP source system to SAP HANA using the SAP HANA studio

    a) Please follow the steps in the downstream steps

Continued on next page

Open Data Provisioning in the ...ick View tab

(if it is not shown go to menu option Help > Quick View).



Step	Action
(25)	Click . Jump to step: SAP HANA Modeler - DATA_PROVISION_EDITOR - SAP HANA Studio

Figure 64: next step

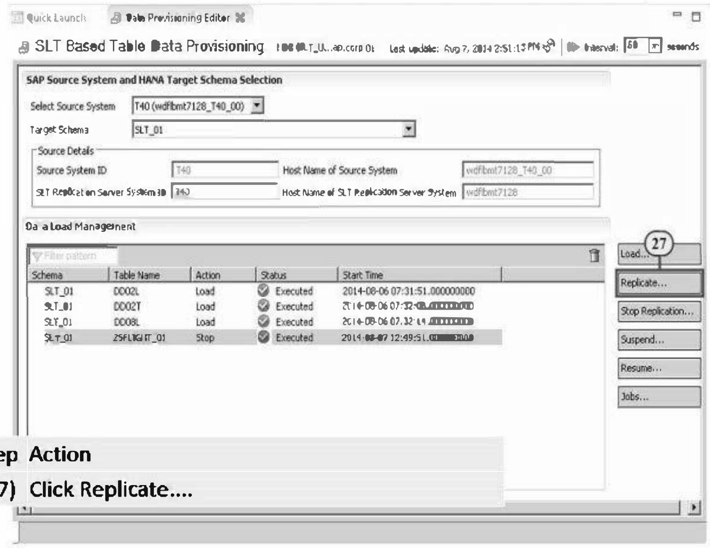
Check the Target Schema.
If necessary, switch to the schema SLT_REPLICATE that was created by the instructor earlier.

Step	Action
(26)	Click Open and switch to target Schema SLT_REPLICATE.

Start the replication of table ZSFLIGHT_## .
(Replace ## with your two digit group number).

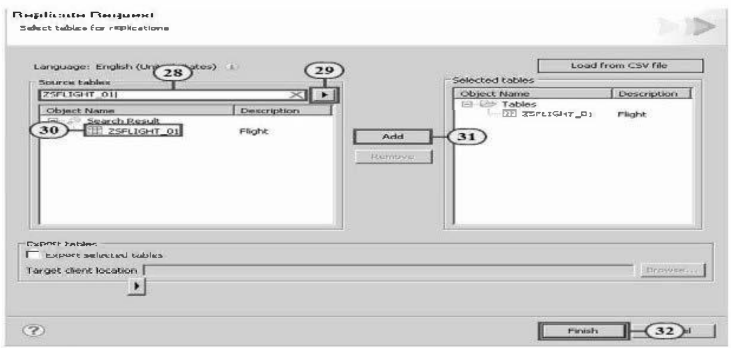
Figure 65: next steps

Continued on next page



Step	Action
(27)	Click Replicate....

Figure 66: next step



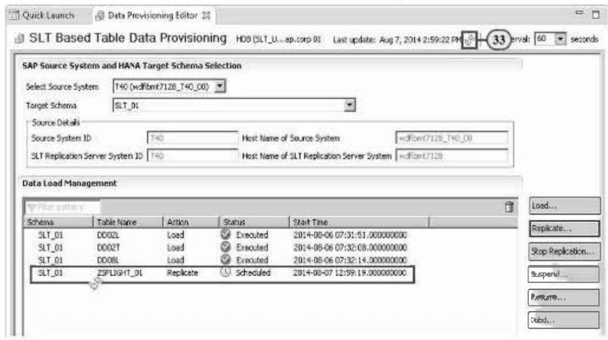
Step	Action
(28)	Enter ZSFLIGHT_## (replace ## with your two digit group number) in the Source tables search field box.
(29)	Click Filter .
(30)	Click ZSFLIGHT_##.
(31)	Click Add.
(32)	Click Finish.

Figure 67: next steps

Continued on next page

3.

a) Please follow the steps in the diagrams



Step	Action
(33)	Click Refresh .

Wait until the table is in action Replication and status In Process.

Figure 68: next step

When the table is in action Replication and status In Process, navigate to table SLT_REPLICATE.ZSFLIGHT_## (replace ## with your two digit group number) and open the data preview.

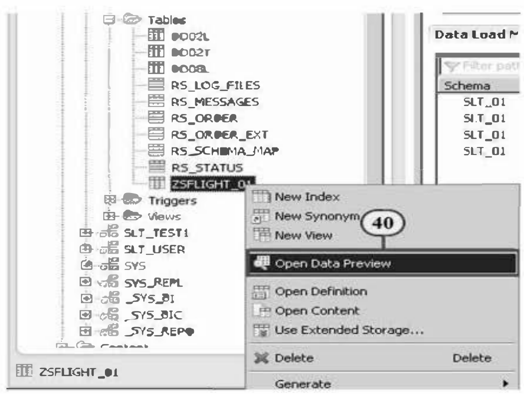


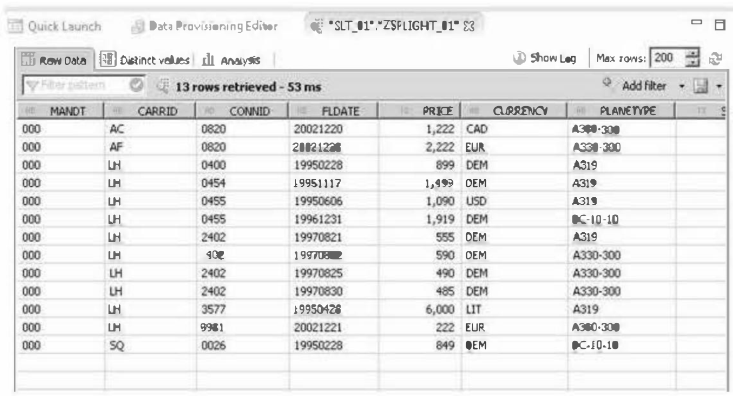
Figure 69: Step 21

Continued on next page

Step	Action
(35)	Open the folder for HDB(SLT_USER) by clicking the Open folder icon.
(36)	Open the folder Catalog by clicking the Open folder icon.
(37)	Open the Schema SLT_REPLICATE by clicking the Open folder icon.
(38)	Open the tables Tables by clicking the Open folder icon.
(39)	Click on ZSFLIGHT_## (replace ## with your two digit group number) with the right mouse button.

If your table does not appear immediately, it may take some time to process it in the training system.
You can refresh by right-click on Tables and choosing Refresh or pressing F5.

Figure 70: next steps

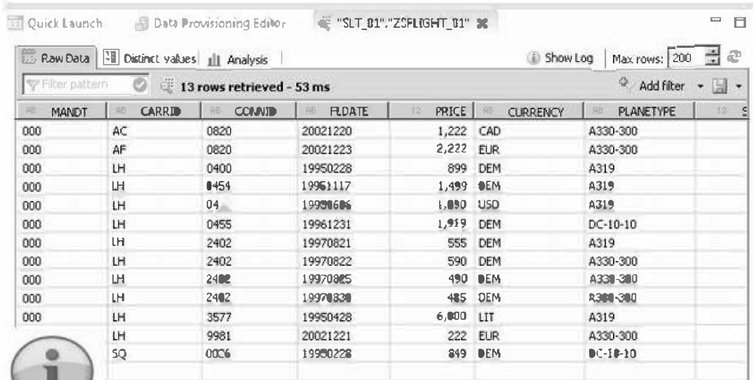


MANDT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
000	AC	0820	20021220	1,222	CAD	A330-300
000	AF	0820	20021220	2,222	EUR	A330-300
000	LH	0400	19950228	899	DEM	A319
000	LH	0454	19951117	1,499	DEM	A319
000	LH	0455	19950606	1,090	USD	A319
000	LH	0455	19961231	1,919	DEM	DC-10-10
000	LH	2402	19970821	555	DEM	A319
000	LH	[illegible]	[illegible]	590	DEM	A330-300
000	LH	2402	19970825	490	DEM	A330-300
000	LH	2402	19970830	485	DEM	A330-300
000	LH	3577	19950428	6,000	LIT	A319
000	LH	[illegible]	20021221	222	EUR	A330-300
000	SQ	0026	19950228	849	DEM	DC-10-10

Figure 71: next step

Continued on next page

Step	Action
(40)	Click the Open Data Preview menu item to execute it.



The table was loaded with records. Any further updates, inserts or deletes will be recorded by SLT and replicated in real time.

Figure 72: Step 40

4.

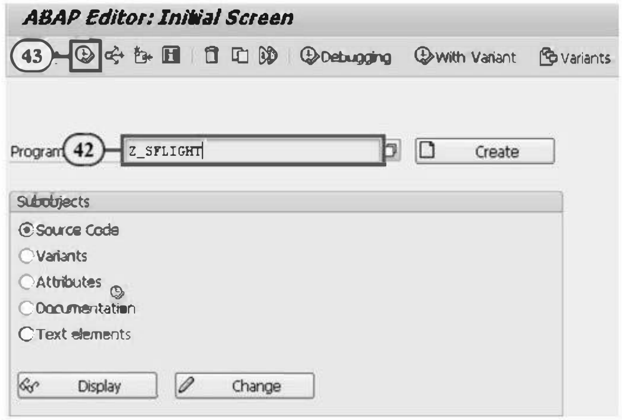
a)

Replicate the new records from the source to the SAP HANA system.
Use transaction SE38 and run the program Z_SFLIGHT to generate new records in your source table ZSFLIGHT_##.
Note that in this system setup, the source system and SLT are running on the same SAP NetWeaver instance.

Step	Action
(41)	**Start transaction SE38.**

Figure 73: Step 41

Continued on next page



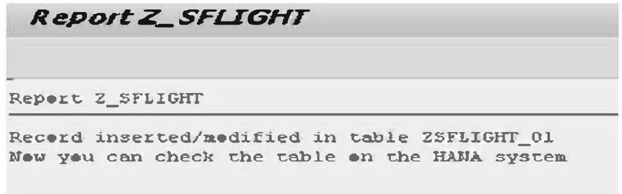
Step	Action
(42)	Enter Z_SFLIGHT.
(43)	Click Execute .

Figure 74: Step 26

Step	Action
(44)	Enter ## (replace ## with your two digit group number in the User Number (P_GROUP)field.
(45)	Click Execute .

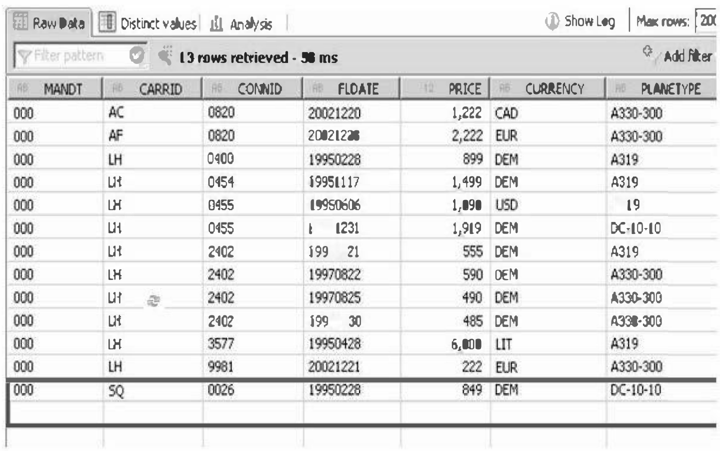
Figure 75: Step 44 and 45

Continued on next page



After you created the new records, switch back to the Data Preview in SAP HANA to verify that the changes have been replicated.

Figure 76: Check the replication



MANDT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
000	AC	0820	20021220	1,222	CAD	A330-300
000	AF	0820	20021220	2,222	EUR	A330-300
000	LH	0400	19950228	899	DEM	A319
000	LH	0454	19951117	1,499	DEM	A319
000	LH	0455	19950606	1,090	USD	19
000	LH	0455	[illegible]1231	1,919	DEM	DC-10-10
000	LH	2402	199 21	555	DEM	A319
000	LH	2402	19970822	590	DEM	A330-300
000	LH	2402	19970825	490	DEM	A330-300
000	LH	2402	199 30	485	DEM	A330-300
000	LH	3577	19950428	6,000	LIT	A319
000	LH	9981	20021221	222	EUR	A330-300
000	SQ	0026	19950228	849	DEM	DC-10-10

Step	Action
(46)	In the SAP HANA Modeler click Refresh .

Figure 77: The result

Lesson Summary

You should now be able to:

- Replicate data to SAP HANA using SLT

Lesson: Overview of transformation possibilities with SLT

Lesson Overview

This lesson will give you an overview about the possibilities of transformation with SLT

Lesson Objectives

After completing this lesson, you will be able to:

- Explain how to change advanced replication settings for certain configurations of SAP LT Replication Server
- Explain the basics of the SLT based transformation concepts
- Know which types of transformation rules exist

Business Example

Your source data needs some transformation before it can be replicated to SAP HANA. You want to understand to what extent can SLT handle data transformations.

Basics of transformation concept

- When the main purpose of SAP HANA is reporting data of selected ERP content, this is generally achieved with a one-to-one replication of ERP tables into the new database.
- Depending on the customer-specific requirements, it is sometimes necessary to filter, change, or extend the original data within the load process into HANA.

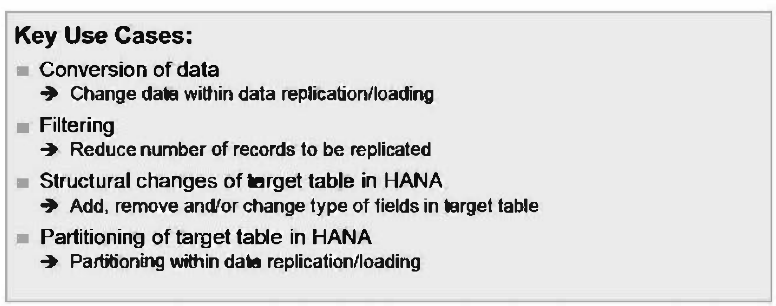


Figure 78: Concept SLT based transformation concepts

The implementation of transformation rules can be achieved in two ways:

- Parameter based
- Event based

A parameter based rule will be executed at each transferred record, it is easy to implement and it can accept up to 3 input parameters. Its logic can be written in a single line of code up to 72 characters long directly in the advanced replication settings transaction, or it can be written in a custom ABAP include, whose name is also specified in the advanced replication settings transaction.

An event based rule requires the knowledge of the SLT internal logic for handling the migration of the data from the source system into the HANA database. The processing logic is outlined in the figure below, data is transferred in blocks and it is possible to enter an event based rules at different stages of the migration steps. The advantage of event based rules over parameter based rules is a greater flexibility and the chance to improve performances. For example if you need to read some master data tables inside your transformation rule and you can do it once by storing the master data information into an internal table, then a parameter based rule would simply lead to a redundant execution of the same selection at each transferred record.

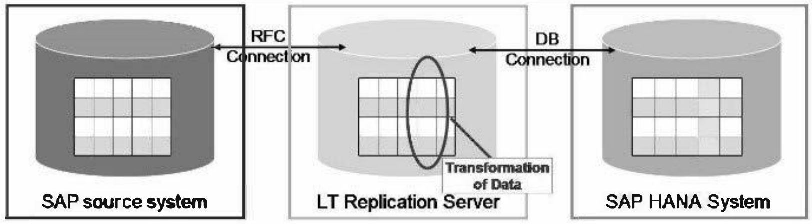
The advanced replication settings for a table are maintained in transaction IUUC_REPL_CONTENT (in SLT up to SP5) or transaction LTRS (from SLT SP6), before starting the replication/load of that specific table.

You can find the detailed steps of how to do this in the new version of SLT in the next exercise.

Transformation of Data

During the replication of the data from the source system into the HANA database via SLT, you may require to apply some transformations on your data. For example you may need to hide some HR related information, e.g. anonymize the names of the employees by entering a sequence of 'X' characters in place of the employee's name. An other example would be the concatenation of the information coming from one field to another. These are only two simple examples of possible transformations applicable during the replication.

Change of Data within Replication by using Transformation Rules



Scenarios
- To make certain fields anonymous → HR reporting
- To fill initial fields e.g. created by change of table stucture
- To convert units or currency and recalculate amounts and values

Options of Implementation
1. Applying Parameter Rules
2. Applying Event-Based Rues

Figure 79: Business Example - Transformation data

- Generally, there are two types of transformations rules, which can be used within transformation
- both has to be created as ABAP include, which can be embedded into replication

Types of Transformation Rules
- **Parameter-based rules**
  - Less flexible than event-based rules
  - Easy to create by using parameters
- **Event-based rules**
  - More flexible than parameter-based rules
  - Knowledge of data processing within the SAP LT Replication Server needed to select the right event for the specific business scenario
  - has access to all fields of a record

Figure 80: SLT based Transformation Concepts - Types of Transformation Rules

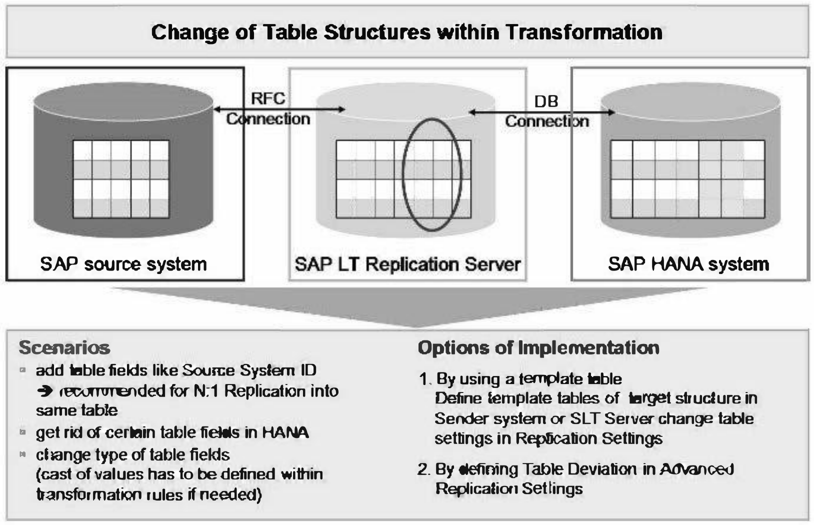


Figure 81: Business Example - Change of (target) table structure

Filtering and selective data Replication

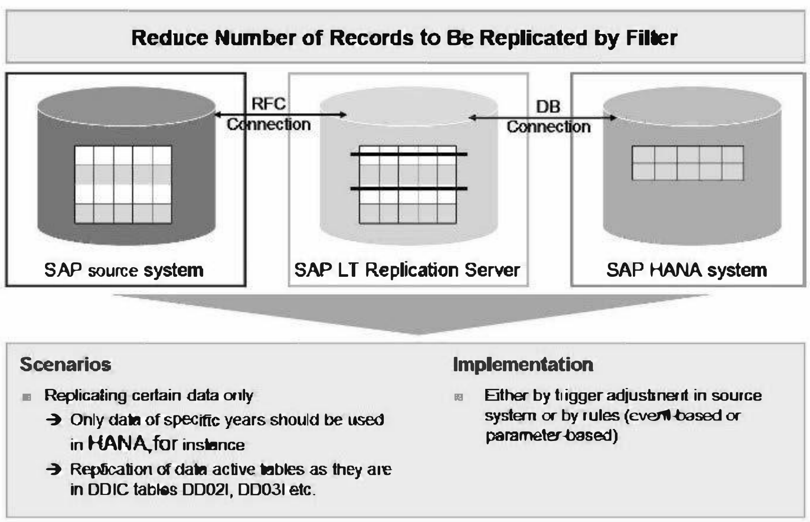


Figure 82: Business Example Selective data replication / filtering

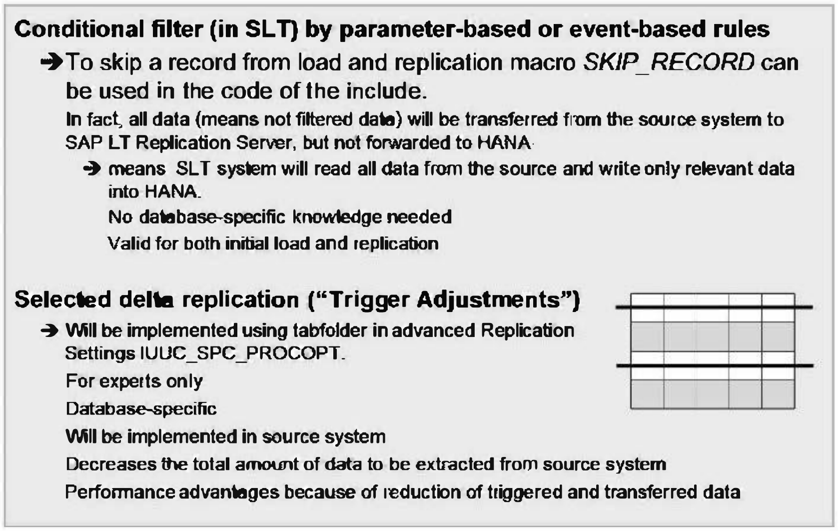


Figure 83: Concept - Filtering Selective data replication

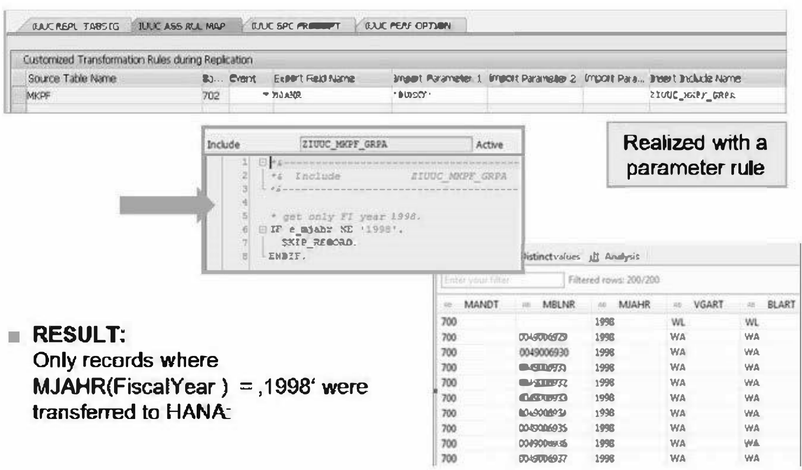
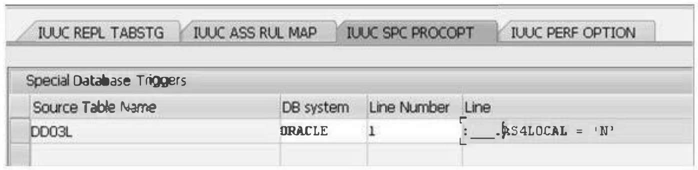


Figure 84: Example: Filtering by Company Code Selective data replication

- In this example, we want to customize the trigger directly in the source system in a way that only changes to the data with AS4LOCAL = 'N' will be recorded.



- **Please note:** trigger filtering will only work for replication phase. If it is required to filter the table during the initial load phase, an additional event (or parameter) filter is also necessary!

Figure 85: Define the Trigger Condition SLT based transformation concepts

Lesson Summary

You should now be able to:

- Explain how to change advanced replication settings for certain configurations of SAP LT Replication Server
- Explain the basics of the SLT based transformation concepts
- Know which types of transformation rules exist

Lesson: Monitoring of Replication

Lesson Overview

The lesson describes the monitoring of replication.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the configuration and monitoring dashboard
- Explain the concept of mass transfer IDs
- Leverage the SLT Cockpit

Business Example

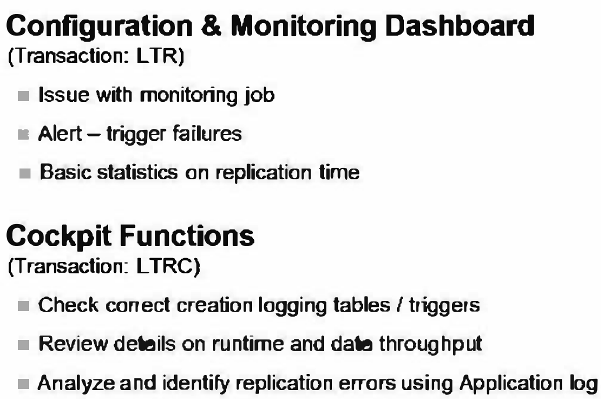
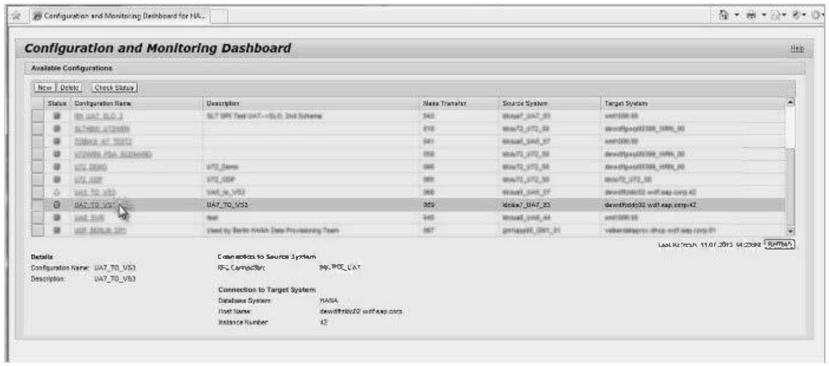


Figure 86: Different Monitoring Capabilities



Within the Configuration and Monitoring Dashboard, click on the configuration for which you would like to display the details.

Figure 87: Status Monitoring – Configuration and Monitoring Dashboard – SAP LT Replication Server

The status can be monitored in the Configuration and Monitoring Dashboard in the LT Replication Server. The detailed monitor can be started by selecting an existing configuration in the *Available Configurations* overview. The status displayed in the overview section is the most severe status derived from the detailed status information.

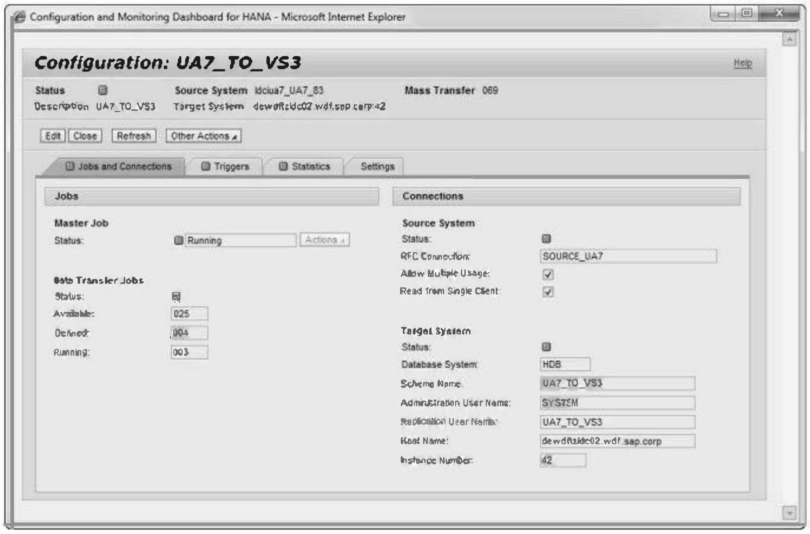


Figure 88: Configuration and Monitoring Dashboard – Details (1/4) – SAP LT Replication Server

The *Jobs and Connections* tab page provides details about the status of the master job, the load jobs, and the connection details of all the involved systems.

If the master job aborts (for example, due to a restart of the LT Replication Server), you can restart the master job by choosing the relevant pushbutton.

The *Available* status shows the total number of available batch tasks in the LT Replication Server; below you can find the number of planned and currently running replication jobs for the current configuration. If there are no tables to be processed, the status might be green even though no replication jobs are running.

In the *Connections* part of the screen, you can view the current connection status to the remote systems (source system and HANA system).

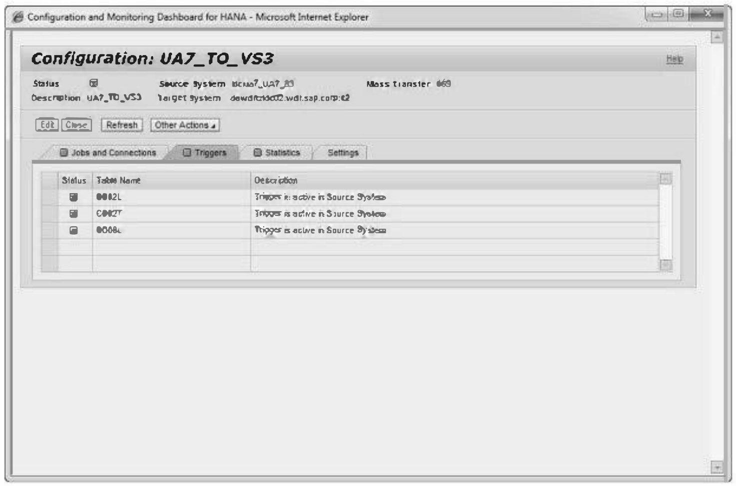


Figure 89: Configuration and Monitoring Dashboard – Details (2/4) – SAP LT Replication Server

On the *Triggers* tab page, you can view the status of the created triggers.

When a new table is scheduled for replication, there will be a warning message that the trigger is not yet created. Note that this message should disappear after a few seconds when the trigger is active.

Note: If you stop a replication, the entry for that table will be completely removed from that list.

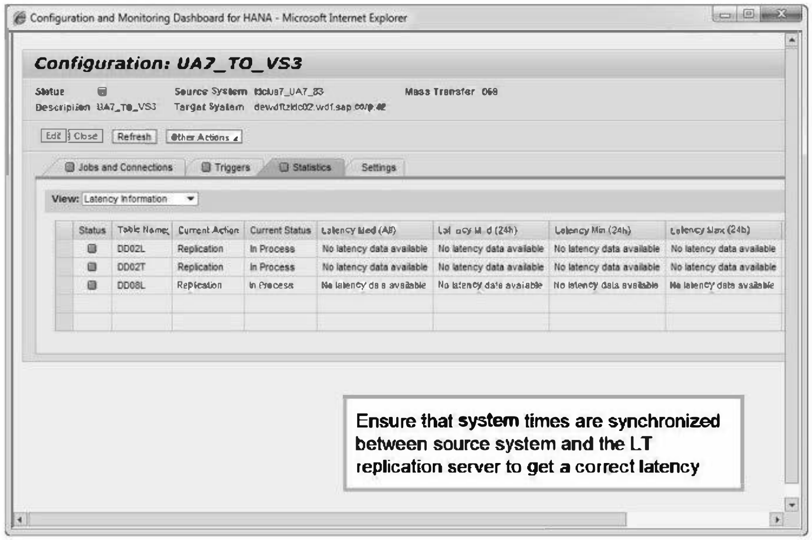


Figure 90: Configuration and Monitoring Dashboard – Details (3/4) – SAP LT Replication Server

On the *Statistics* tab page, you can view the current status of all relevant tables of the selected configuration. You can view the *Current Action* and the *Current Status*.

For all tables started in replication mode, and when the initial replication of all data is completed, a latency is calculated and also displayed here. The latency is the time difference between the change timestamp on the source system and the commit timestamp in the HANA system.

There are different latency figures calculated for each replicated table:

- **Latency Med (all):** median of all latency figures
- **Latency Med (24h):** median of the latency figures from the last 24 hours
- **Latency Min (24h):** minimum latency within last 24 hours
- **Latency Max (24h):** maximum latency within last 24 hours
- **Last Replication:** laps of time since last replication

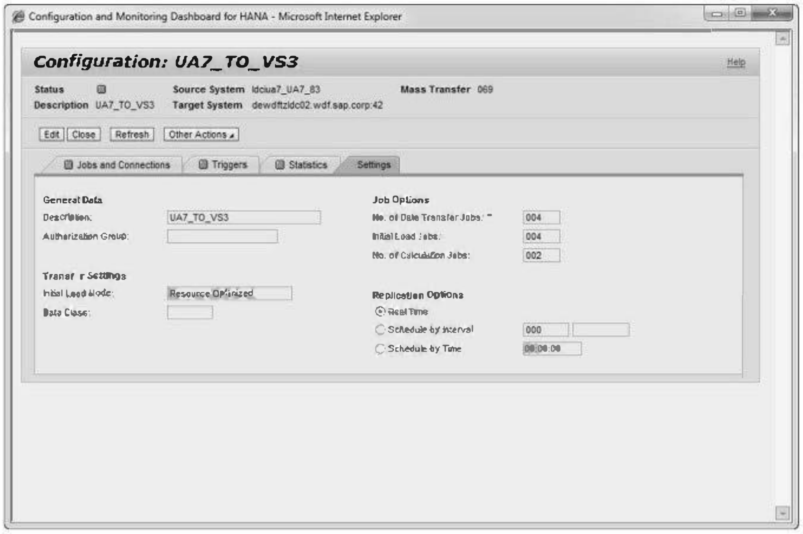


Figure 91: Configuration and Monitoring Dashboard – Details (4/4) – SAP LT Replication Server

On the *Settings* tab page, the data entered when creating the configuration are displayed. If you switch to the *Edit* mode, some properties can be changed.

The *Number of Data Transfer Jobs* can be increased or decreased. On the *Settings* tab, the maximum number of jobs used for the initial load can be modified. The number of *Initial Load Jobs* defines how many replication jobs can be used for initial load. Therefore, this number cannot be higher than the total number of data transfer jobs. We recommend that you choose the number of *Initial Load Jobs* one job less than the number of total jobs, so that there is at least one job reserved for delta replication. The number of *Calculation Jobs*, used for access plans calculations, can also be adjusted here.

You can also switch the replication mode between Real Time replication and Scheduled replication.

The mass transfer ID of a configuration schema is displayed as follows:

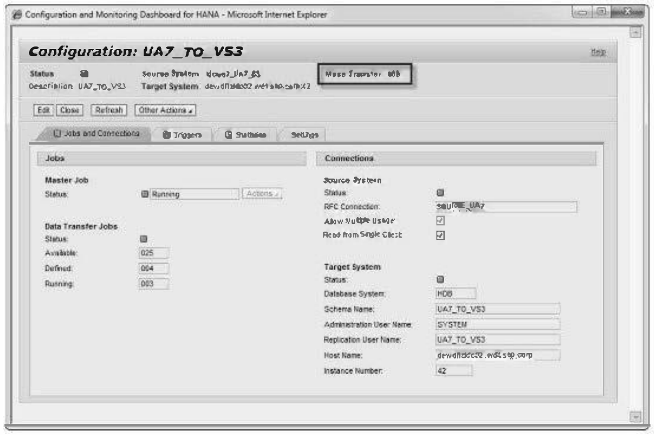


Figure 92: Mass Transfer IDs

When you start the LT Replication Server Cockpit (LTRC) you have to enter the mass transfer ID or select it form the dropdown-list.

When creating a new configuration schema in the LT Replication Server, a mass transfer ID (MT_ID) is created and assigned to this configuration. The numbering of the mass transfer IDs will continuously increase on the LT Replication Server when new configuration schemas are defined.

The SLT Cockpit is available via transaction *LTRC*

- Administration → General information
- Processing Steps → Advanced manual handling of replication steps.
- Table Overview → Information about replicated tables, log tables, and so on
- Data Transfer Monitor → Information about runtime objects
- Application Logs → Log information
- Load Statistics → Statistics on transferred data
- Expert Functions → Collection of Expert Functions

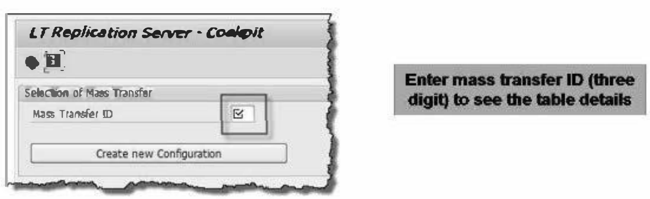


Figure 93: SLT Cockpit

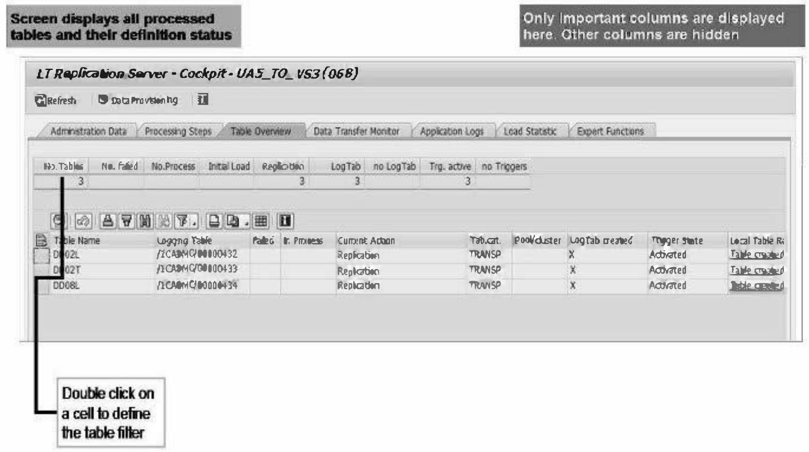


Figure 94: SLT Cockpit – Table Overview

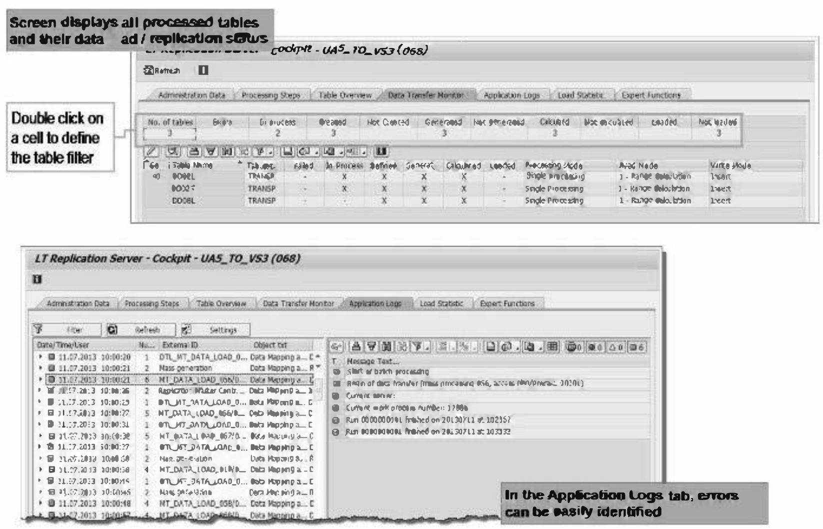
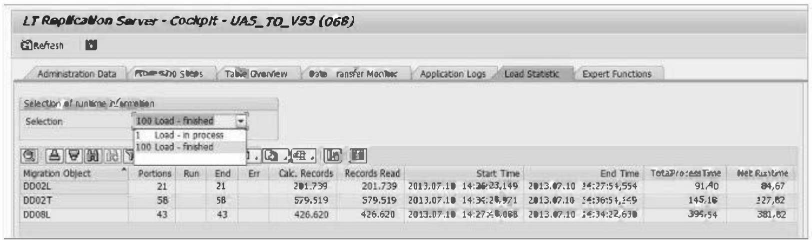


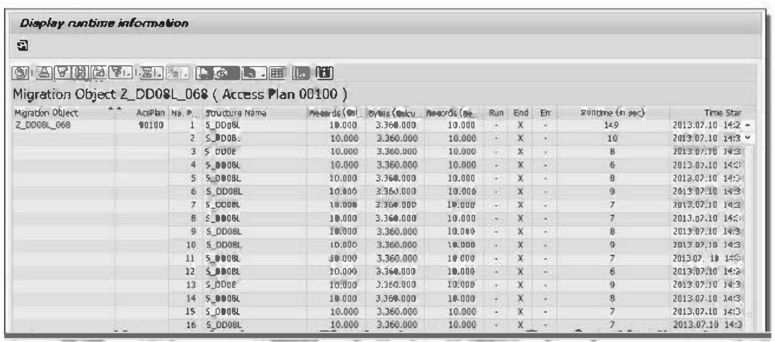
Figure 95: SLT Cockpit – Data Transfer Monitor



The statistics tab can show alternatively the statistics for the finished tables or for the tables being replicated.

Figure 96: SLT Cockpit – Statistics

You can enter the detail view by double click on a table name



In this screen, the individual status for each load portion is displayed. Also the total runtime and also the runtime for each individual step (for example to read the data) is displayed here and gives a first indication about potential bottlenecks during the data transfer.

Figure 97: Data Transfer Monitor – Runtime Information for Load – Details

Lesson Summary

You should now be able to:

- Describe the configuration and monitoring dashboard
- Explain the concept of mass transfer IDs
- Leverage the SLT Cockpit

Unit Summary

You should now be able to:

- Explain the positioning of SAP LT Replication Server
- Describe the key concepts and features
- Describe the key steps to create a configuration
- Replicate data to SAP HANA using SLT
- Explain how to change advanced replication settings for certain configurations of SAP LT Replication Server
- Explain the basics of the SLT based transformation concepts
- Know which types of transformation rules exist
- Describe the configuration and monitoring dashboard
- Explain the concept of mass transfer IDs
- Leverage the SLT Cockpit

Unit 3

SAP Data Services

Unit Overview

Data Service is a powerful tool for data provisioning within in a Business Objects dominated IT- landscape

In this unit you will learn how you can use this tool for data provisioning.

Unit Objectives

After completing this unit, you will be able to:

- Explain Data Services capabilities with SAP HANA

Unit Contents

Lesson: SAP Data Services and SAP HANA

Lesson Overview

This lesson will guide you through loading data into SAP HANA using SAP Data Services.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain Data Services capabilities with SAP HANA

Business Example

You want to load data into SAP HANA which does not require real-time updating, and you want to perform a large number of transformations to the data. You have decided that SAP Data Services is the right tool.

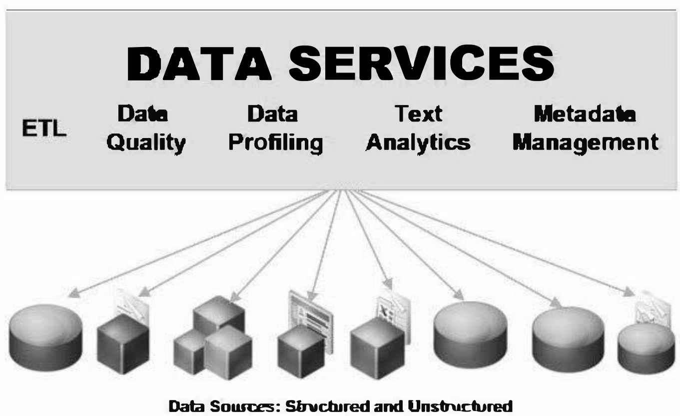


Figure 98: Solution: One-Stop Solution for Information Management

There are many components to Data Services.

These include

- **Data integration** (ETL) - extract, harmonize and combine any data from disparate sources
- **Data quality** - fix bad data, add missing fields, add geo information, find best records, de-duplicate
- **Text analytics** - identify key entities and sentiments in unstructured data, social networks etc. and combine with structured data
- **Data quality** - fix bad data, add missing fields, add geo information, find best records, de-duplicate
- **Data profiling** - discover how clean is your data
- **Metadata Management** - manage entire enterprise meta data in one repository, data lineage, metapedia (glossary of business terms)

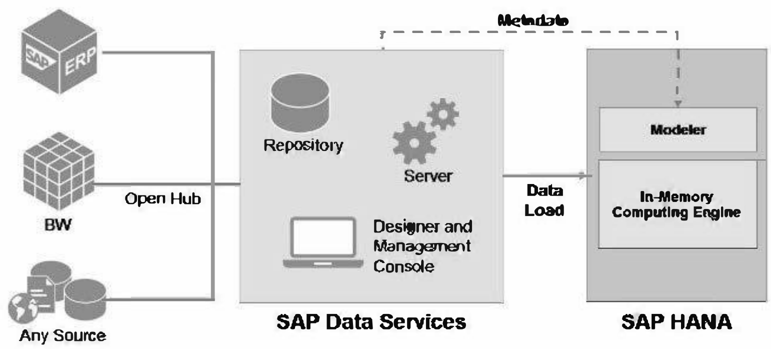


Figure 99: SAP Data Services - SAP HANA Extract, transform, and load data quickly

SAP Data Services enable you to gain deeper insight with a single, trusted view by accessing and integrating structured and unstructured data from data sources across your enterprise. With data integrator you can,

Extract, transform, and load data into a data warehouse to create a complete view

Unlock meaning from unstructured documents with native text data processing

The data integrator capabilities can be used with many of our data stores, including SAP HANA, SAP NetWeaver BW, SAP Rapid Marts, and Sybase IQ.

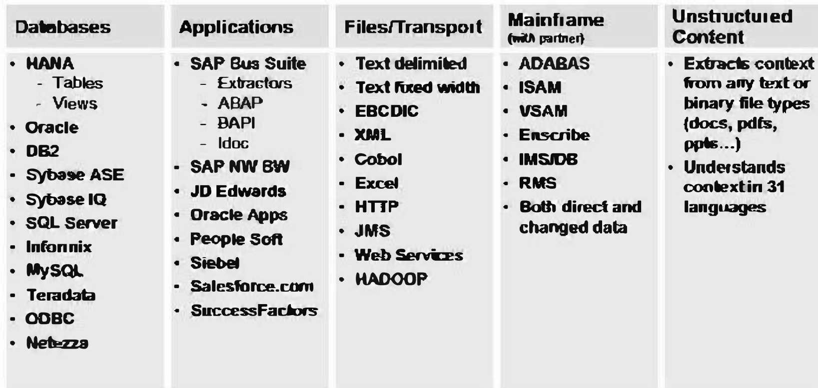


Figure 100: The strength of Data Services

A key strength of Data Services is the long history of support for a huge number of data sources including:

Databases,

Applications

Files,

Legacy sources ,

Unstructured content

Plus it is possible to create you own custom adaptor for non standard sources

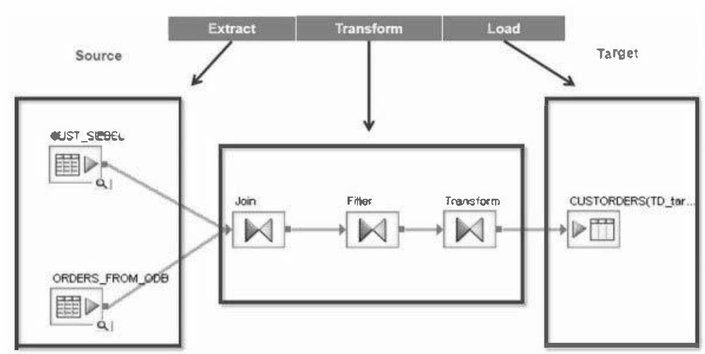


Figure 101: ETL with Data Services

Traditionally, Data Services performs the transformations of data in the Data Services engine and then loads the results to the data target.

This is known as extraction, transformation, loading (ETL). However with SAP HANA we switch things around to improve performance.

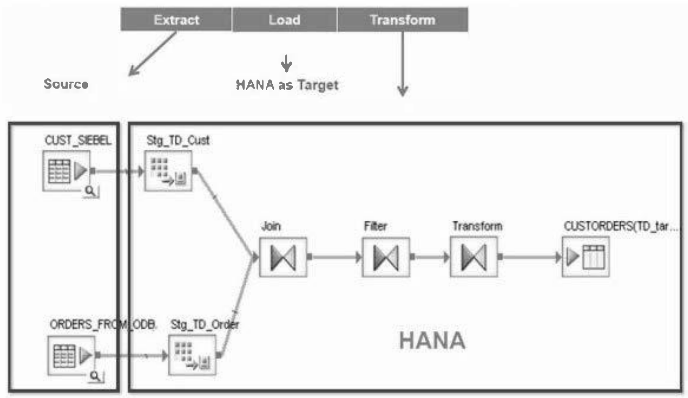


Figure 102: SAP HANA ensures for best performance

When loading data to SAP HANA we can exploit the power of HANA to take care of the transformations. Data Services stages the source data then pushes down the transformation processing to SAP HANA. This means superior performance compared to Data Services in a non-HANA scenario. This means we now have a new process of extraction, loading, transformation (ELT). It is critical to remember we are still defining the job in Data Services so at the job design level things look the same.

But under the covers the processing is very different with SAP HANA doing the **heavy lifting**.

Additional Sources Support with Data Services

- HANA stored procedures, calculation, attribute, and analytical views as a Source
- Optimized Hadoop extraction and loading
- SuccessFactors – source and target
- Binary files (PDF, Doc) and 31 additional languages supported for Text data processing

Loading Performance

- 50% improvement in loading flat file and database to HANA (via 'DS-HANA pipe')
- Increased E-L-T and push down optimizations
- Sybase IQ – support for bulk updates and binary format data transfer
- 10x improvement in BW extract and load to HANA, Sybase
- Parallelized reading from Business Content Extractors

Modeling, Metadata, and Repository

- New Eclipse DS Workbench for very fast, simplified data movement into HANA
- Full support for HANA views for metadata impact and lineage analysis
- HANA as a repository for Data Services and Information Steward

Figure 103: Enhancements in version 4.1 and beyond

Data Services enjoyed a large number of enhancements in version 4.1 and beyond.

Take a look at the items in this list to see how many features were developed with SAP HANA in mind.

Remember, Data Services remains a valid ETL/ELT tool for loading data to SAP HANA where customers have complex requirements for transforming, cleansing and enhancing data from multiple sources and a batch drive approach is required.

But customers should also look at the new EIM capabilities (SDI and SDQ) built into SAP HANA that may satisfy simpler ETL/ELT requirements and this of course means a simpler landscape.

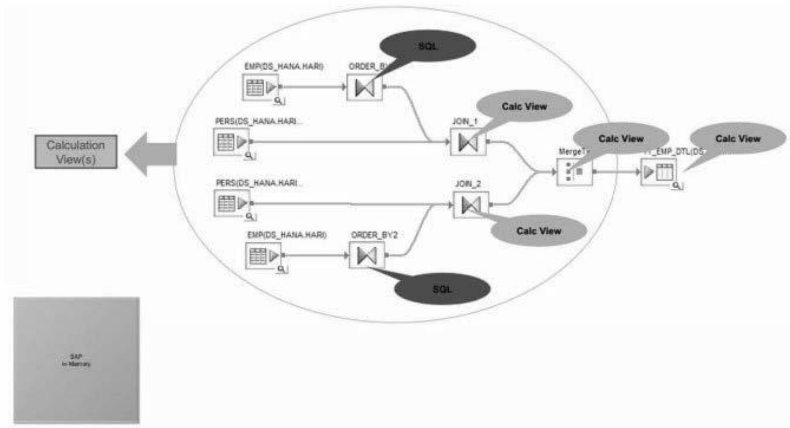


Figure 104: Pushing down processing to SAP HANA

Data Services jobs push down processing to SAP HANA for improved performance wherever possible.

For example, if SAP HANA is the data source, we push down the filtering and also any joins, lookups etc. to SAP HANA. If SAP HANA is the data target we push down the processing to the target SAP HANA. Data Services figures out best how to push down based on each data flow scenario.

Also for complex transforms we also push down processing to SAP HANA using stored procedures.

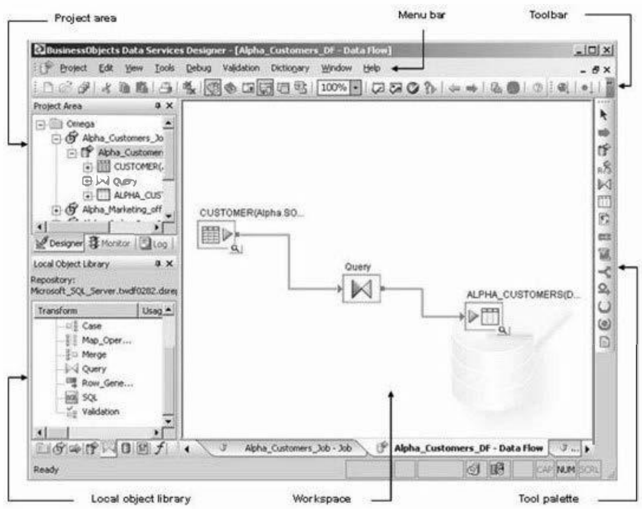


Figure 105: Data Services Designer interface

Loading jobs are created using the Data Services Designer. It is a **Windows client** and utilises a highly graphical environment for ease of use.

Job are created by dragging objects from the repositories to the canvas.

Advanced debug, audit and trace features are available.

Meta data is imported from source and target systems to save having to create this manually. This is a huge time saver and ensures synchronization.

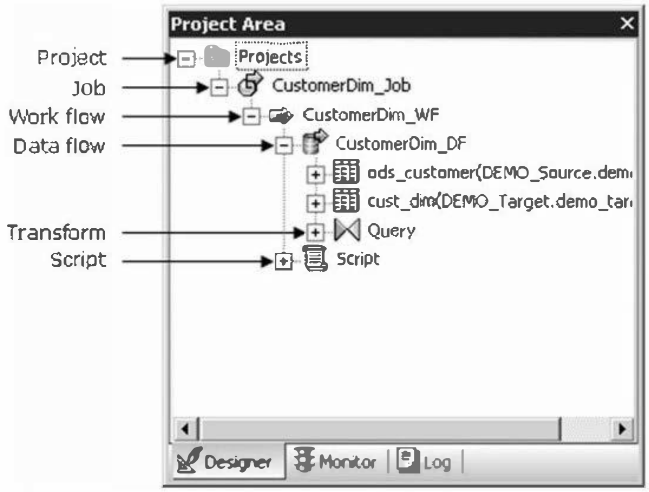


Figure 106: Data Services objects

Project: like a folder, this is used to collect all objects for a complete loading scenario (e.g. load weekly and daily sales master data and transaction data) Job: these are the executable, schedulable objects than contain either workflows, data flows or both (e.g. daily sales load, weekly sales load). Workflows: these are sub jobs and are optional but help to break down very large jobs into smaller jobs (e.g. load master data, load transaction data) Dataflow : the smallest flow object which is where you define what actually happens in the ETL steps (e.g. load new customers, adjust discount codes, load returns) Scripts : optional code to fine tune logic in flow

Not shown: Datastores: definitions of sources and target databases and applications File Formats : Flat files, XML, XL, text documents).

Most of the objects created can be reused in other flows.

- **SAP HANA repository support**
- **SAP HANA performance improvements**
- **Bulk updates enhancement**
- **Support for stored procedures**
- **Data streaming in ABAP data flows**
- **SAP table reader in regular data flows**
- **Parallel reading from business content extractors**
- **New ABAP functions for improved security**

Figure 107:

- **SAP HANA repository support**

  Data Services now supports the SAP HANA database as a repository host. You are no longer required to use a separate database to host the Data Services repository.

- **SAP HANA performance improvements**

  Data Services automatically detects the SAP HANA target table type and updates the table accordingly for optimal performance.

- **Bulk updates enhancement**

  For improved performance when using changed-data capture or auto-correct load, Data Services now uses a transparent staging table for bulk loading to targets.

- **Support for stored procedures**

  Data Services now supports SAP HANA stored procedures. Scalar data types are supported for input and output parameters. Procedures can be called from a script or a Query transform as a new function call.

- **Data streaming in ABAP data flows**

  This version includes a new data transfer method option RFC, which lets you stream data from the source SAP system directly to the Data Services data flow process using RFC. This enhancement eliminates the need for intermediate data files and also provides the ability to adjust the data transfer package size.

- **SAP table reader in regular data flows**

  Previously, reading large data volumes could result in an out-of-memory error on the SAP source side. The Data Services SAP table reader in regular data flows can now fetch data in batch from an SAP system. This new implementation allows the reader to process large volumes of data and mitigates out-of-memory errors.

- **Parallel reading from business content extractors**

  Data Services now supports multithreading for SAP extractors for improved performance.

Exercise 2: Replication and Transformation with SAP Data Services

Exercise Objectives

After completing this exercise, you will be able to:

- Develop an SAP Data Services job to load data to SAP HANA

Business Example

You need to merge data from an SAP ERP source system with data from a flat file and load the results to an SAP HANA table..

Task:

Create a Data Services job to merge data from an SAP table and and a flat file.

START > All programs > Data services 4.2 > Data Sevices Designer

SYSTEM	**wdflbmt5074.wdf.sap.corp**
USER:	train-##
PASS	train-##
AUTHENTICATION	Enterprise
Please press L●G●N	
Repository	DSREPO##
Press OK	

Result

1. Launch SAP Data Services Designer and connect to your repository.

a) In the Local Object Library pane (bottom left) click on tab Datastores

Continued on next page

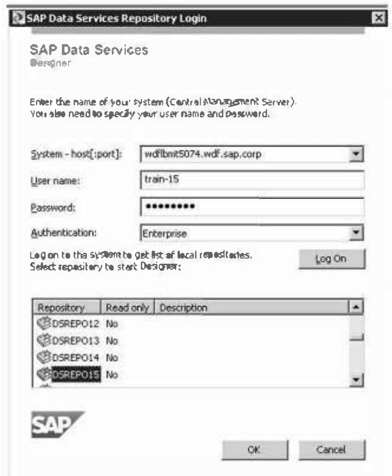


Figure 108: .

2. Create a new datastore for the SAP source system and import the metadata from ABAP table SFLIGHT_##

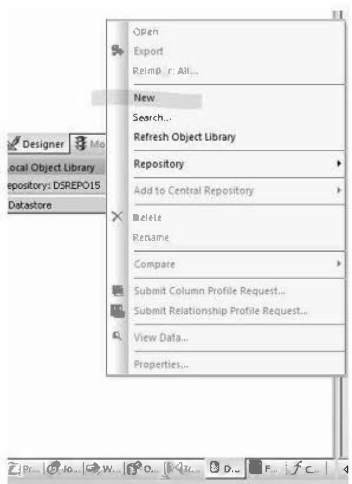


Figure 109: .

a) Right-click in the empty space and chose New

Continued on next page

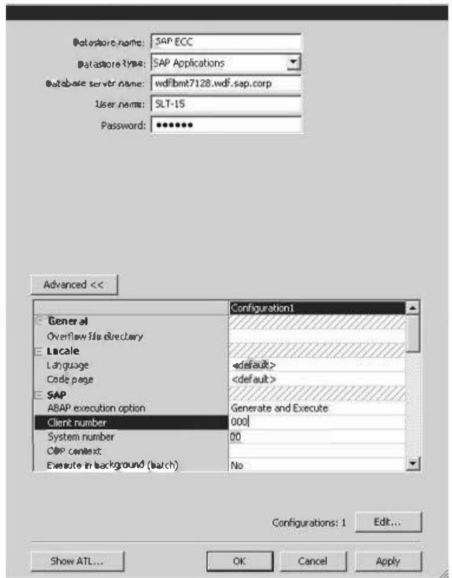


Figure 110: .

b) Enter the following details:

Datastore name:	SAP_ECC
Datastore type	SAP Applications
Database server name:	wdflbmt7128.wdf.sap.corp
User Name	SLT-##
Password:	INITIAL
	Press **ADVANCED**
Client number:	000
System Number:	00 (this is the default)
	Press **OK**

Continued on next page

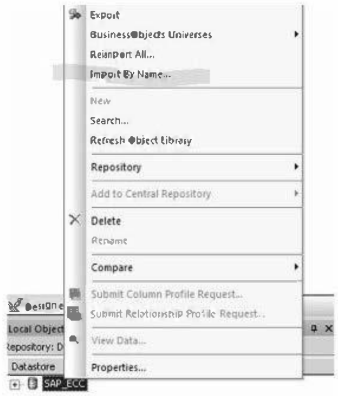


Figure 111: .

c) Right click on SAP_ECC datastore and select Import by Name

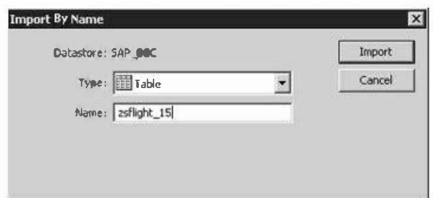


Figure 112:

d) Complete the pop-up entries:

Type: Table

Name: zsflight_##

Press **Import**

3. Create a new datastore for the SAP HANA target system.

Continued on next page

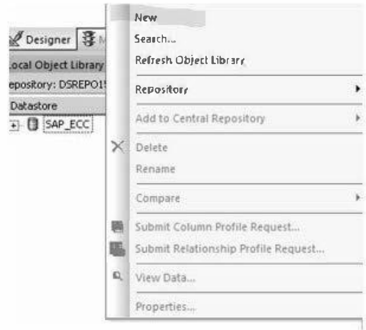


Figure 113: New datastore .

a) Anywhere in the empty space right-click **New**

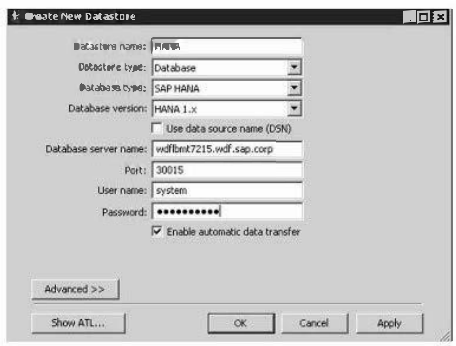


Figure 114: .

b) Enter the following details:

Datastore name:	HANA
Datastore type:	Database
Database type:	SAP HANA
Database version:	HANA 1.x
Database server name:	wdflbmt7215.wdf.sap.corp
Port:	30015
User Name:	system
Password:	W3lcome123

Continued on next page

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

	Press **OK**
Client number:	000
System Number:	00

4. Create a new file format to define the meta data for the flat file.

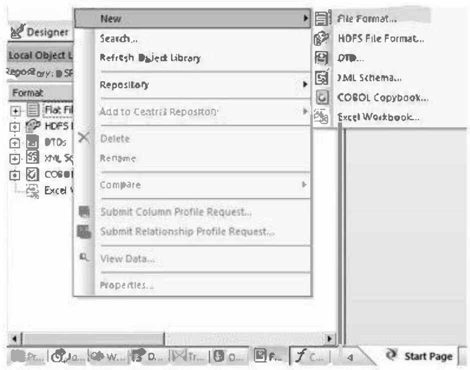


Figure 115: .

a) Click on tab **Formats** (bottom of pane)

bi) Right-click in empty space and select **New > File Format**

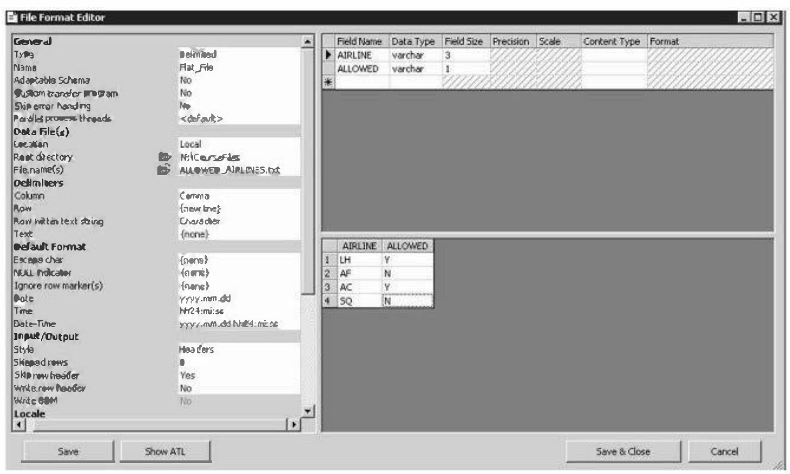


Figure 116: .

Continued on next page

c) Leave all default values and enter only the following:

Name:	Flat_File
Root directory	N:\ (press the yellow icon to find this drive)
File names(s):	ALLOWED_AIRLINES.txt (press the yellow icon to select this folder)
Skip row header:	Yes

IN the meta data table (top right) change the Field Size for AIRLINE from 2 to 3

Press **Save and Close**

5. Create a new project AIRLINES

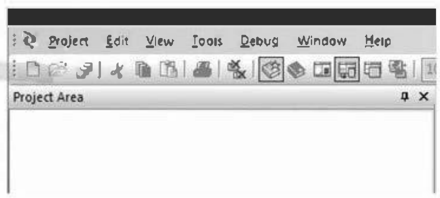


Figure 117: .

a) Click on New Project

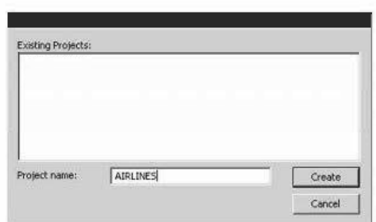


Figure 118: .

b Enter the project name : AIRLINES Press **Create**

6. Create a new batch job AIRLINES_JOB

Continued on next page

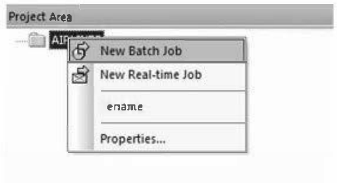


Figure 119: .

c) Right-click on the new project and choose **New Batch Job**

7. Insert a dataflow AIRLINE_DF in your job.

Figure 120: .

a) Change the name of the job to airlines_job

.

Figure 121: .

b) From the vertical toolbar on the far right of the Designer click once on Dataflow icon (3rd from top) and then click again in the centre of the screen to add a Dataflow to the job

c) Change the name of the Dataflow to **airline_DF**

Continued on next page

d) Double click on the new Dataflow to open the Dataflow editor

8. Insert the SAP table as a source to the dataflow.

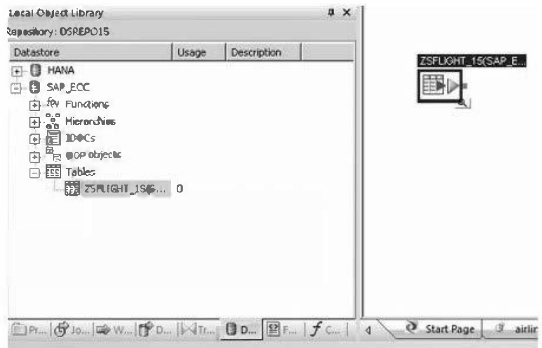


Figure 122: .

a) Select the Datastores tab and expand SAP_ECC > Tables so you see table ZSFLIGHT_##

b) Drag the table ZSFLIGHT_## onto the canvas

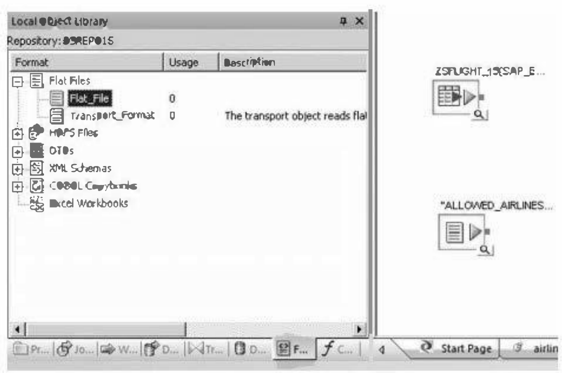


Figure 123: .

c) Switch to the Formats tab and drag Flat_File t the canvas. Choose Source when prompted as you drop the file

9. Insert a query transform into the flow after the data sources and connect them.

Continued on next page

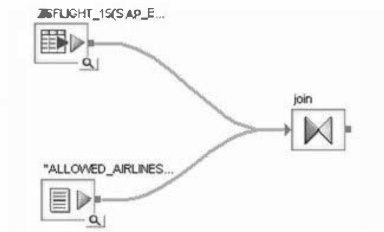


Figure 124: .

d) Drag a Query transform from the vertical toolbar to the canvas. Change the name to join and connect the data sources to the Query transform.

10. Define the column mapping for the SAP table and the flat file to the query transform.

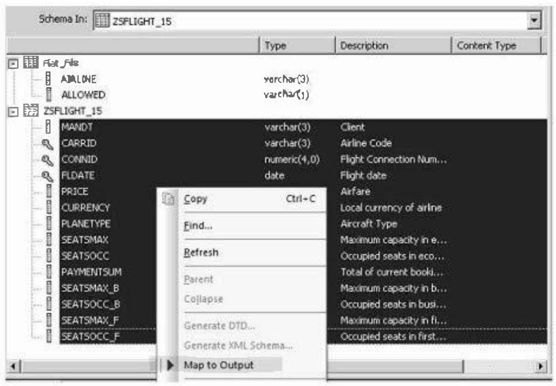


Figure 125: .

a) Double click on the Query transform icon to open the mapping editor and highlight all columns from the ZSFLIGHT input schema (not the table name!) then right-click and choose Map to Output

Continued on next page

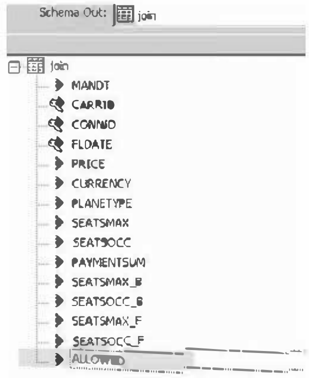


Figure 126: .

b) From the input schema Flat_File, drag the column Allowed to the output (drop in the lower empty space)

11. Define the inner join condition for the two data sources.

Figure 127: .

a) Select the **Where** tab and from the input schema drag the **CARRID** column to the Where pane, then type an equals sign with space before and after, then drag AIRLINE column to complete the join argument

ZSFLIGHT_15.CARRID = Flat_File.AIRLINE

12. Insert a validation transform into the flow after the query transform.

Continued on next page

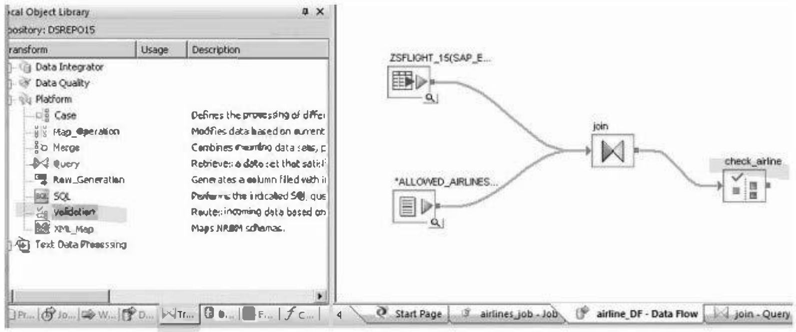


Figure 128: .

a) Navigate back to the Dataflow view (use the arrow back or close the window or click on the Dataflow in the project area).

Select the **Transforms tab** and expand Platform node then drag the **Validation** transform to the canvas to the right of the join node and connect the join transform to the validation transform. Change the name of the validation transform to **check _airline**

13. Configure the validation rules so that we can check if the airline is approved for the bookings made by our team members.

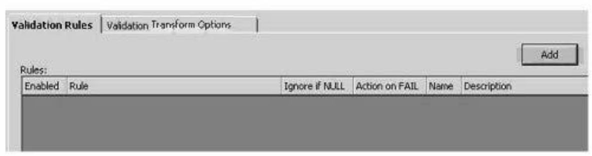


Figure 129: .

a) Double-click on the validation transform to open the editor and press the Add button to create new rule

Continued on next page

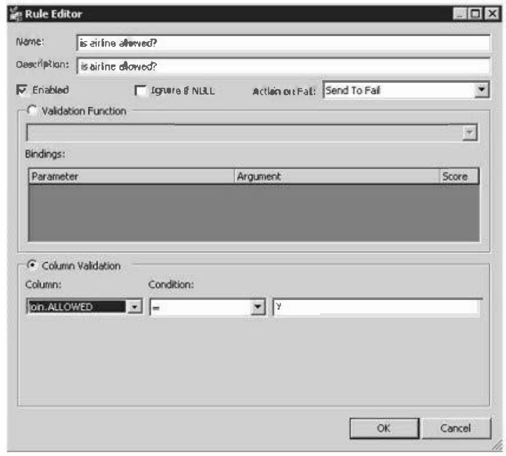


Figure 130: .

Enter the rule as follows:

Name: is airline allowed?

Description: is airline allowed?

•

Select radio button **Column Validation**

Select from the column drop down: **join.ALLOWED**

Condition : =

Value: **Y (upper case)**

Press **OK**

14. Insert a template table to catch the valid records.

Continued on next page

Figure 131:

Figure 132: .

Connect the validation transform to the template table and at the prompt select Pass

15. Insert a template table to catch the invalid records.

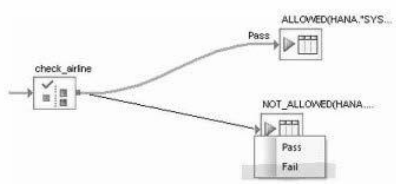


Figure 133: .

Add a second template table to the flow. At the prompt enter the following details:

Table name:	not_allowed
In datastore:	HANA
Owner name:	SYSTEM

Continued on next page

Click **OK**

Then connect the nodes this time using the option **Fail**

16. Change the flat file location to point to a copy of the file on the job server so that it can be found when the batch job runs.

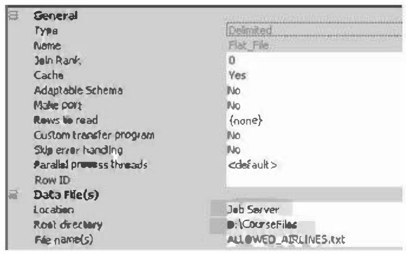


Figure 134: .

Before you can run the job you need to change the location of the file in the flow so that the job can find the flat file. Double-click on the Allowed Airlines source and change the settings:

Location:	Job Server
Root directory:	D:\CourseFiles

Note: When our job runs the flat file must be located on the job server, the job cannot read the local file server.

But equally we cannot import flat file metadata automatically from a flat file located in the job server. That is why we used the version of the file in the local folder earlier so we didn't have to type manually the meta data. We put a copy of the flat file on the local server and also on the job server.

17. Run the job

Figure 135: .

Continued on next page

From the **Project Area** right-click on the job and choose **Execute** to start the job At the next prompt accept all default entries and press **OK** .

Hint: Wait for the job to finish (**20 seconds**) and check the last line of the log shows it finished successfully.

18. Check the results by locating the newly created tables in SAP HANA and make sure they both contain data.

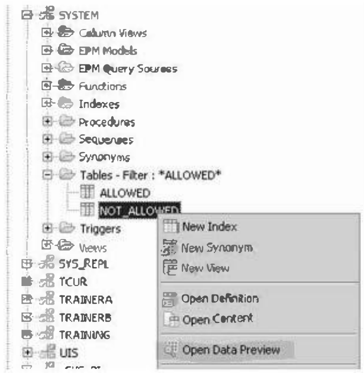


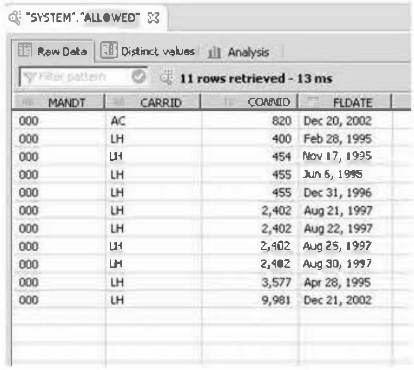
Figure 136:

In the HANA Studio locate the two newly created tables

CATALOG > SYSTEM > TABLES

Right click on both tables and choose **Open Data Preview** to view the records

Continued on next page

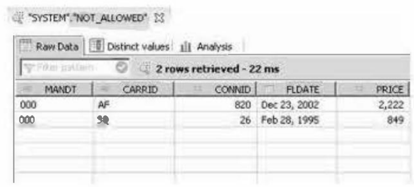


"SYSTEM"."ALLOWED"

Raw Data | Distinct values | Analysis

11 rows retrieved - 13 ms

MANDT	CARRID	CONNID	FLDATE
000	AC	820	Dec 20, 2002
000	LH	400	Feb 28, 1995
000	LH	454	Nov 17, 1995
000	LH	455	Jun 6, 1995
000	LH	455	Dec 31, 1996
000	LH	2,402	Aug 21, 1997
000	LH	2,402	Aug 22, 1997
000	LH	2,402	Aug 25, 1997
000	LH	2,402	Aug 30, 1997
000	LH	3,577	Apr 28, 1995
000	LH	9,981	Dec 21, 2002



"SYSTEM"."NOT_ALLOWED"

Raw Data | Distinct values | Analysis

2 rows retrieved - 22 ms

MANDT	CARRID	CONNID	FLDATE	PRICE
000	AF	820	Dec 23, 2002	2,222
000	SQ	26	Feb 28, 1995	849

Figure 137: .

You should see both HANA tables have records

One table contains records for airlines that are not allowed (2 records), the other table contains records that are allowed (11 records)

Solution 2: Replication and Transformation with SAP Data Services

Task:

Create a Data Services job to merge data from an SAP table and and a flat file.

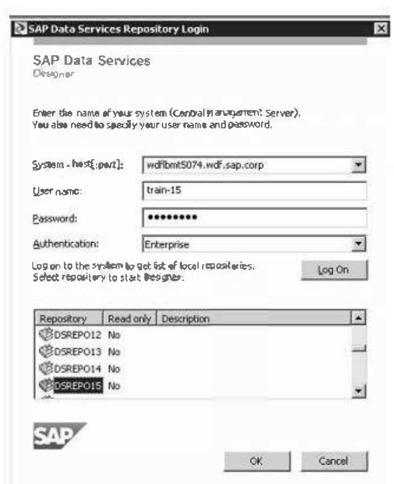
START > All programs > Data services 4.2 > Data Sevices Designer

SYSTEM	**wdflbmt5074.wdf.sap.corp**
USER:	train-##
PASS	train-##
AUTHENTICATION	Enterprise
Please press LOGON	
Repository	DSREPO##
Press OK	

Result

1. Launch SAP Data Services Designer and connect to your repository.

a) In the Local Object Library pane (bottom left) click on tab Datastores



Continued on next page

Figure 138: .

2. Create a new datastore for the SAP source system and import the metadata from ABAP table SFLIGHT_##

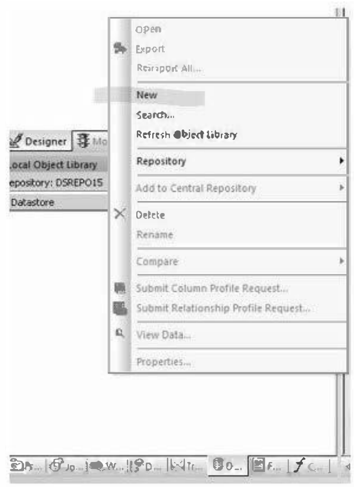


Figure 139: .

a) Right-click in the empty space and chose New

Continued on next page

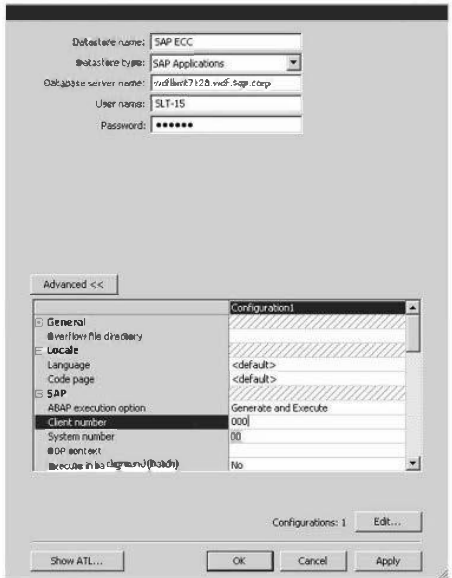


Figure 140: .

b) Enter the following details:

Datastore name:	SAP_ECC
Datastore type	SAP Applications
Database server name:	wdflbmt7128.wdf.sap.corp
User Name	SLT-##
Password:	INITIAL
	Press **ADVANCED**
Client number:	000
System Number:	00 (this is the default)
	Press **OK**

Continued on next page

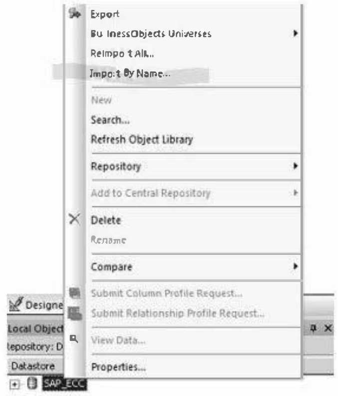


Figure 141: .

c) Right click on SAP_ECC datastore and select Import by Name

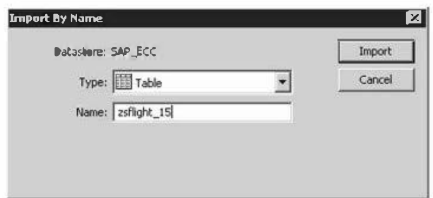


Figure 142:

d) Complete the pop-up entries:

Type: Table

Name: zsflight_##

Press **Import**

3. Create a new datastore for the SAP HANA target system.

Continued on next page

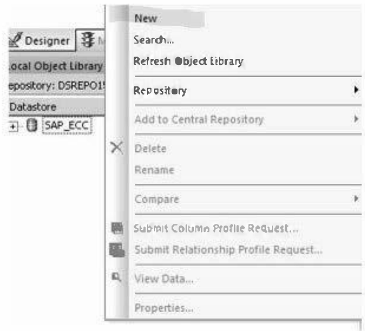


Figure 143: New datastore .

a) Anywhere in the empty space right-click **New**

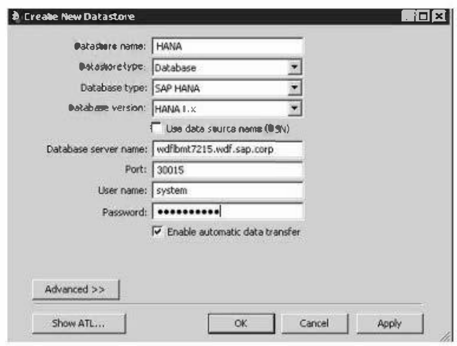


Figure 144: .

b) Enter the following details:

Datastore name:	HANA
Datastore type:	Database
Database type:	SAP HANA
Database version:	HANA 1.x
Database server name:	wdflbmt7215.wdf.sap.corp
Port:	30015
User Name:	system
Password:	W3lcome123

Continued on next page

	Press ●K
Client number:	000
System Number:	00

4. Create a new file format to define the meta data for the flat file.

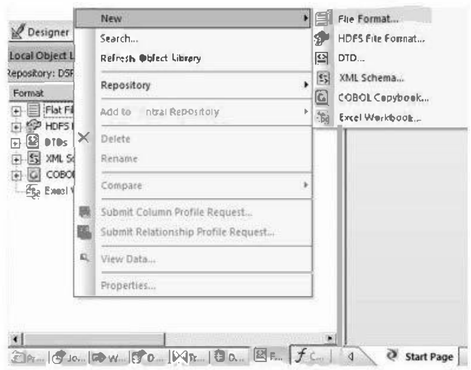


Figure 145: .

a) Click on tab **Formats** (bottom of pane)

bi) Right-click in empty space and select **New** > **File F●rmat**

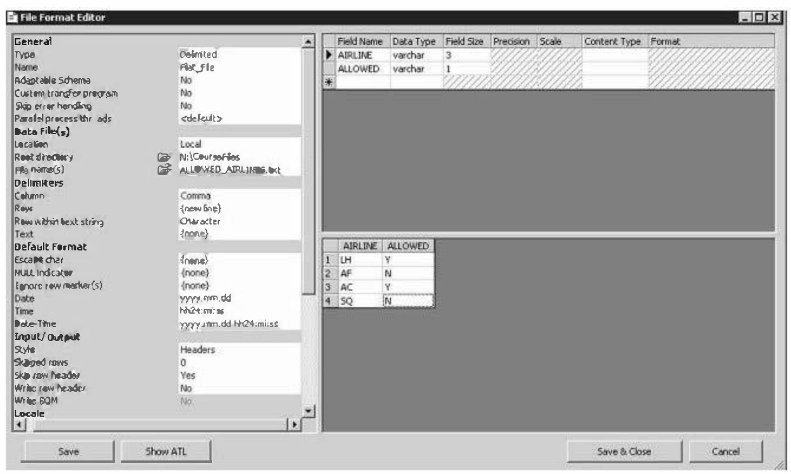


Figure 146: .

Continued on next page

c) Leave all default values and enter only the following:

Name:	Flat_File
Root directory	N:\ (press the yellow icon to find this drive)
File names(s):	ALLOWED_AIRLINES.txt (press the yellow icon to select this folder)
Skip row header:	Yes

IN the meta data table (top right) change the Field Size for AIRLINE from 2 to 3

Press **Save and Close**

5. Create a new project AIRLINES

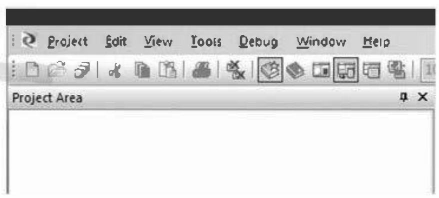


Figure 147: .

a) Click on New Project

Figure 148: .

b Enter the project name : AIRLINES Press **Create**

6. Create a new batch job AIRLINES_JOB

Continued on next page

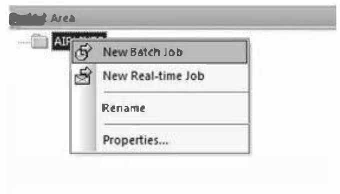


Figure 149: .

c) Right-click on the new project and choose **New Batch Job**

7. Insert a dataflow AIRLINE_DF in your job.

Figure 150: .

a) Change the name of the job to airlines_job

.

Figure 151: .

b) From the vertical toolbar on the far right of the Designer click once on Dataflow icon (3rd from top) and then click again in the centre of the screen to add a Dataflow to the job

c) Change the name of the Dataflow to **airline_DF**

Continued on next page

d) Double click on the new Dataflow to open the Dataflow editor

8. Insert the SAP table as a source to the dataflow.

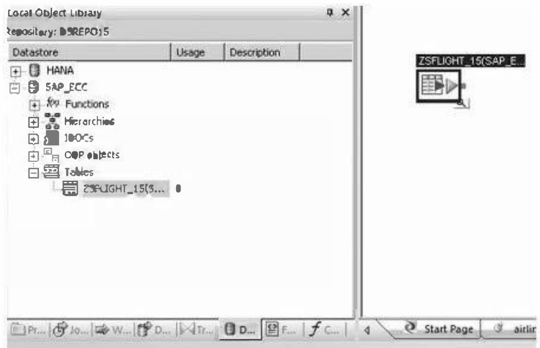


Figure 152: .

a) Select the Datastores tab and expand SAP_ECC > Tables so you see table ZSFLIGHT_##

b) Drag the table ZSFLIGHT_## onto the canvas

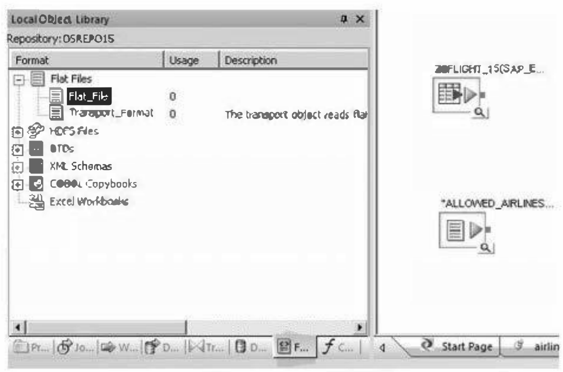


Figure 153: .

c) Switch to the Formats tab and drag Flat_File t the canvas. Choose Source when prompted as you drop the file

9. Insert a query transform into the flow after the data sources and connect them.

Continued on next page

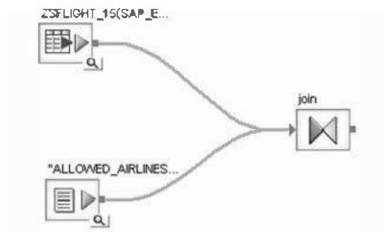


Figure 154: .

d) Drag a Query transform from the vertical toolbar to the canvas. Change the name to join and connect the data sources to the Query transform.

10. Define the column mapping for the SAP table and the flat file to the query transform.

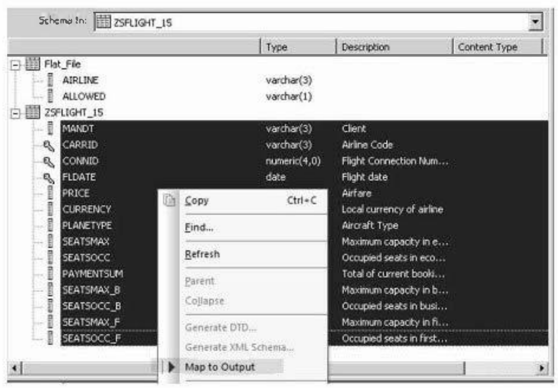


Figure 155: .

a) Double click on the Query transform icon to open the mapping editor and highlight all columns from the ZSFLIGHT input schema (not the table name!) then right-click and choose Map to Output

Continued on next page

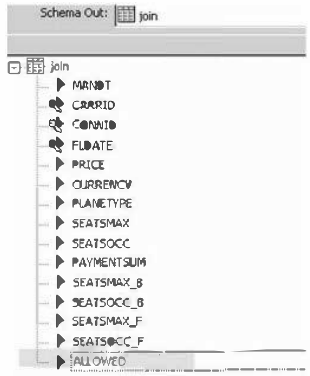


Figure 156: .

b) From the input schema Flat_File, drag the column Allowed to the output (drop in the lower empty space)

11. Define the inner join condition for the two data sources.

Figure 157: .

a) Select the **Where** tab and from the input schema drag the **CARRID** column to the Where pane, then type an equals sign with space before and after, then drag AIRLINE column to complete the join argument

ZSFLIGHT_15.CARRID = Flat_File.AIRLINE

12. Insert a validation transform into the flow after the query transform.

Continued on next page

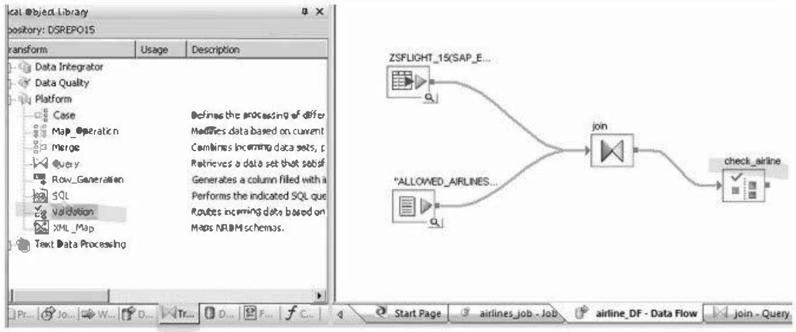


Figure 158: .

a) Navigate back to the Dataflow view (use the arrow back or close the window or click on the Dataflow in the project area).

Select the **Transforms tab** and expand Platform node then drag the **Validation transform** to the canvas to the right of the join node and connect the join transform to the validation transform. Change the name of the validation transform to **check _airline**

13. Configure the validation rules so that we can check if the airline is approved for the bookings made by our team members.

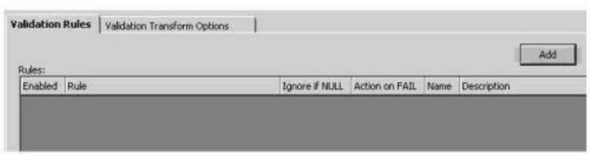


Figure 159: .

a) Double-click on the validation transform to open the editor and press the Add button to create new rule

Continued on next page

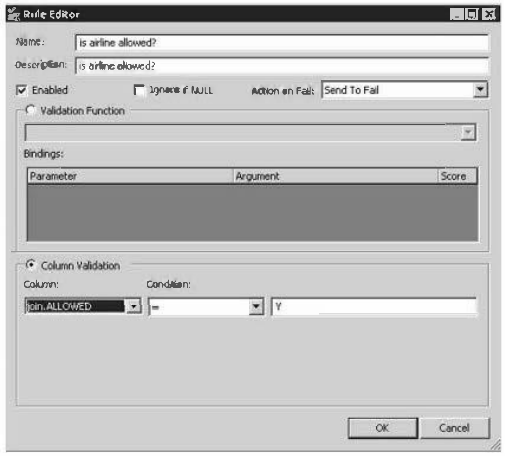


Figure 160: .

Enter the rule as follows:

Name: is airline allowed?

Description: is airline allowed?

•

Select radio button **Column Validation**

Select from the column drop down: **join.ALLOWED**

Condition : =

Value: **Y (upper case)**

Press **OK**

14. Insert a template table to catch the valid records.

Continued on next page

Figure 161:

Figure 162: .

Connect the validation transform to the template table and at the prompt select Pass

15. Insert a template table to catch the invalid records.

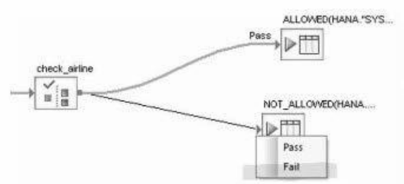


Figure 163: .

Add a second template table to the flow. At the prompt enter the following details:

Table name:	not_allowed
In datastore:	HANA
Owner name:	SYSTEM

Continued on next page

Click **OK**

Then connect the nodes this time using the option **Fail**

16. Change the flat file location to point to a copy of the file on the job server so that it can be found when the batch job runs.

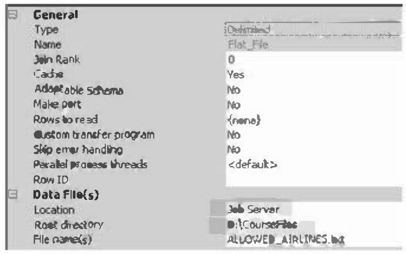


Figure 164: .

Before you can run the job you need to change the location of the file in the flow so that the job can find the flat file. Double-click on the Allowed Airlines source and change the settings:

Location:	Job Server
Root directory:	D:\CourseFiles

Note: When our job runs the flat file must be located on the job server, the job cannot read the local file server.

But equally we cannot import flat file metadata automatically from a flat file located in the job server. That is why we used the version of the file in the local folder earlier so we didn't have to type manually the meta data. We put a copy of the flat file on the local server and also on the job server.

17. Run the job

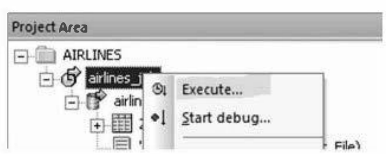


Figure 165: .

Continued on next page

From the **Project Area** right-click on the job and choose **Execute** to start the job At the next prompt accept all default entries and press **OK** .

Hint: Wait for the job to finish (**20 seconds**) and check the last line of the log shows it finished successfully.

18. Check the results by locating the newly created tables in SAP HANA and make sure they both contain data.

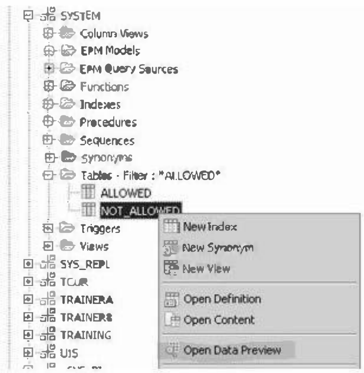


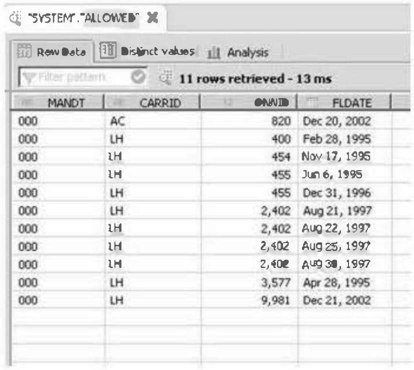
Figure 166:

In the HANA Studio locate the two newly created tables

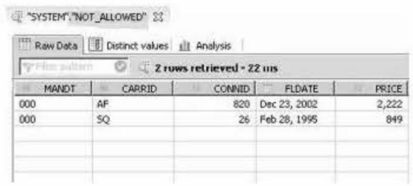
CATALOG > SYSTEM > TABLES

Right click on both tables and choose **Open Data Preview** to view the records

Continued on next page



MANDT	CARRID	CONNID	FLDATE
000	AC	820	Dec 20, 2002
000	LH	400	Feb 28, 1995
000	LH	454	Nov 17, 1995
000	LH	455	Jun 6, 1995
000	LH	455	Dec 31, 1996
000	LH	2,402	Aug 21, 1997
000	LH	2,402	Aug 22, 1997
000	LH	2,402	Aug 25, 1997
000	LH	2,402	Aug 30, 1997
000	LH	3,577	Apr 28, 1995
000	LH	9,981	Dec 21, 2002



MANDT	CARRID	CONNID	FLDATE	PRICE
000	AF	820	Dec 23, 2002	2,222
000	SQ	26	Feb 28, 1995	849

Figure 167: .

You should see both HANA tables have records

One table contains records for airlines that are not allowed (2 records), the other table contains records that are allowed (11 records)

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Lesson Summary

You should now be able to:

- Explain Data Services capabilities with SAP HANA

Unit Summary

You should now be able to:

- Explain Data Services capabilities with SAP HANA

Unit 4

Smart Data Access (SDA)

Unit Overview

Smart Data Access is a powerful new tool to address integration and cost reduction of a landscape, and a foundation technology underlying the SAP Real-Time Data Platform (RTDP).

Unit Objectives

After completing this unit, you will be able to:

- Understand the Smart Data Access concepts
- Know when to use Smart Data Access
- Know how to implement Smart Data Access
- Build data models using virtual tables.

Unit Contents

Lesson: Introduction to Smart Data Access

Lesson Overview

●n this lesson you will be introduced to the Smart data Access functionality in SAP HANA. This is an important building block for understanding how SAP gets their various database products to work together, and how their Big Data solutions work.

Lesson Objectives

After completing this lesson, you will be able to:

- Understand the Smart Data Access concepts
- Know when to use Smart Data Access
- Know how to implement Smart Data Access
- Build data models using virtual tables.

Business Example

A retail company does not want to load all their retail data into SAP HANA as most of their reports are only using the last 6 weeks of data. But for some reports they would like e.g. to compare this year's sales to last year's sales. So they do not want to archive the data, but want it available in their reports. In the case of reports on older data they do not expect the normal fast HANA reporting times. You can store the older data in a database like SAP IQ, and transparently to the user fetch it from there in case a report requires this older data. For all reports on newer data, the SAP HANA system still delivers fast reporting times. Smart Data Access provides such capabilities. And it is standard functionality built into SAP HANA.

Introduction to Smart Data Access

Smarter Data Virtualization

SAP HANA smart data access brings in-memory computing to data virtualization providing a non-disruptive intelligent architecture to provision enterprise-wide data from heterogeneous systems such as Hadoop, Teradata, other SAP HANA systems, SAP ASE and SAP IQ. Experience data like never before. In an "always-on" and rapidly changing world Big Data is enabling new customer interactions and insights, new experiences, and new business processes. Businesses are rethinking how they work and imagining what was previously impossible. SAP understands customers current business practices and has the expertise and technologies such as in-memory and cloud to exploit Big Data.

Think about an orchestra playing. Harmony and coordinated cooperation is needed to succeed. Each player has a piece and contributes to the whole. SAP HANA is the conductor and soloist. Here we look at how SAP HANA integrates with Hadoop and other databases providing smart data access and discusses how organizations can become more competitive by leverage insight from all their data to support real-time business.

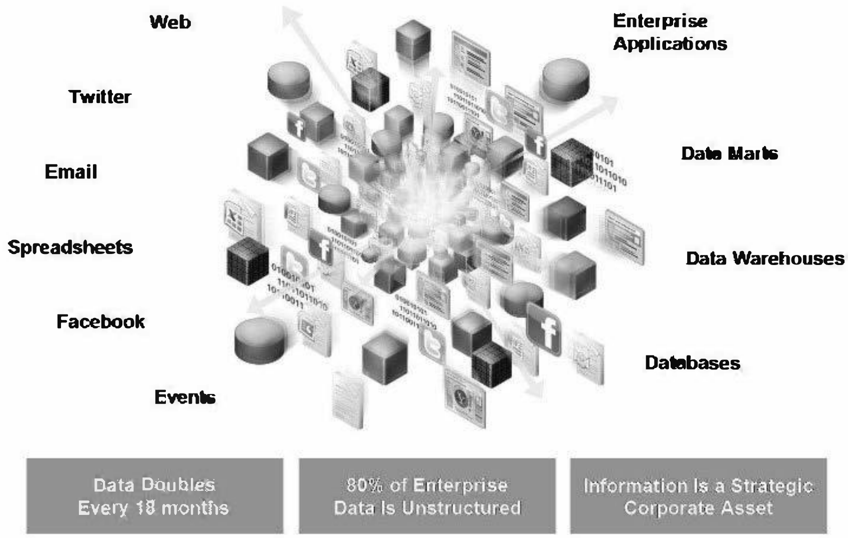


Figure 168: Information Explosion is Driving Change

The Information Explosion is driving change everywhere. Information is growing at an exponential rate driven by traditional data sources like spreadsheets, databases, data warehouses, enterprise applications. However, new types of data from unstructured data sources such as Twitter, Facebook, email are eclipsing the volume of data being created in the traditional systems. In fact data is doubling every 18 months (according to IDC) and most of the data in an enterprise, 80% to be exact, is coming from unstructured sources.

We all recognize that information can be a strategic corporate asset giving us insights to what products are selling, what customers are buying, and what customers are saying about us in social forums like support blogs, Twitter, and Facebook.

To harness the power of information requires a strategy for amassing, storing, analyzing and gleaning informational insight. Customers need a foundation that can deal with the data deluge. SAP HANA with smart data access capability provides this foundation.

Figure 169: Diverse Technologies

Enterprises will have many tools from different vendors supporting data management initiatives for ETL, data quality, data profiling, metadata management, and text analytics. They desire to simplify their IT environment with a single foundation to deliver data services across your entire enterprise (for current or new projects). Such a foundation needs to be open to support all data sources and targets, scalable to handle small and extreme data volumes, and reusable so that they can build once and reuse for other projects.

Such an integrated Big Data foundation can be the standard for delivering all of the critical information management capabilities across the enterprise. As a result, customers can significantly gain greater IT efficiency and deliver maximum business effectiveness.

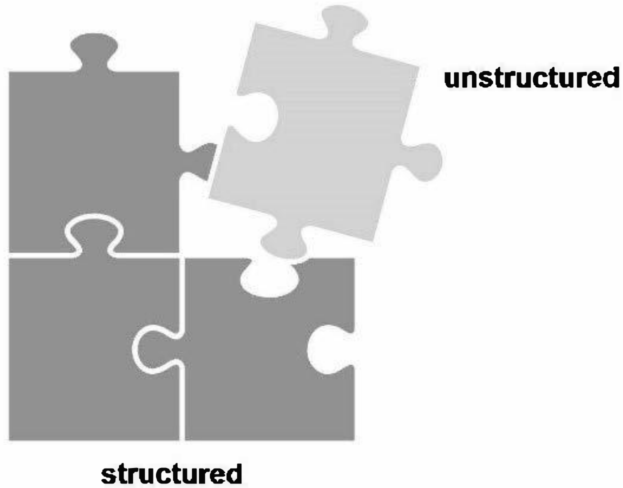


Figure 170: In This Era of Big Data, What If You Could Combine structured and unstructured data for new insights....

In This Era of Big Data, What If You Could Combine structured and unstructured data for new insights....

Nowadays almost every company is faced with managing vast amounts of data from a variety data sources. These range from:

- Traditional databases that mange transactional data from business applications, to
- Enterprise data warehouses that facilitate deep analysis to
- Hadoop that stores large volumes of mostly unstructured data that can be mined for newer insights

What if you could, take advantage of new technologies such as Hadoop, MapReduce, and text data processing to

- Mine large volumes of unstructured data located on low-cost storage
- Extract and load relevant data from these large data sets into an in-memory database
- Combine unstructured with structured data for insights never seen before?

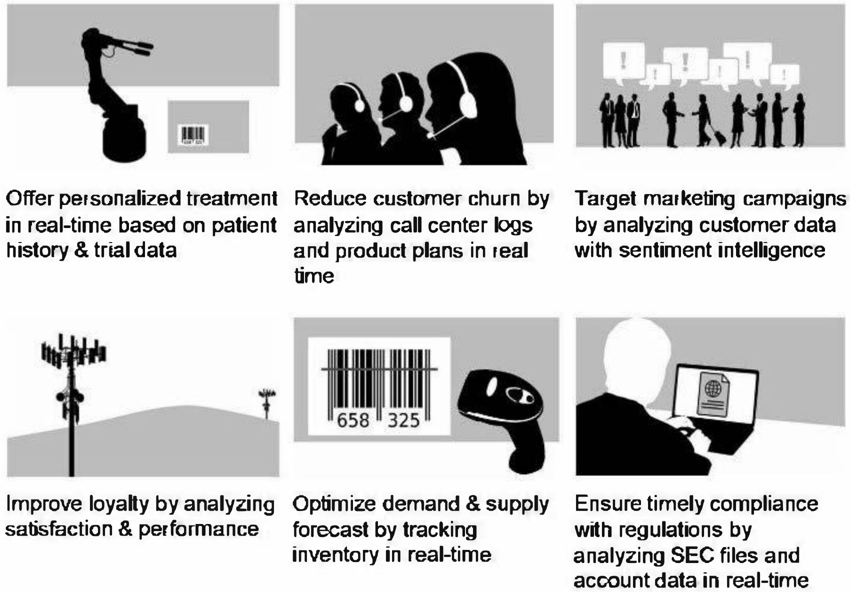


Figure 171: ... And analyze it in Real-Time to improve business processes?

What of you could analyze all that data in real-time to improve business processes?

SAP helps enterprises combine new and relevant data types with structured data to improve business processes.

The heart and soul of most organizations is the transactional data used by mission-critical applications to run the day-by-day operations of the business. Competitive organizations can no longer maintain the "status quo" – by only analyzing traditional, structured data stored in databases and data warehouses.
Today, organizations must sift through large volumes of new data – from Web logs, social networks, barcodes, and sensors – and identify relevant nuggets of information that are pertinent to their business. Then, savvy organizations combine relevant information with structured data to gain new insights never seen before in time to enable real-time business. For example, enterprise organizations are taking advantage of these new insights to improve:

- Marketing, sales, service
- Brand sentiment, campaign performance, 360 view of customer, online customer habits, customer loyalty, customer churn, competitor analysis, product recommendations, product profitability, service opportunity

Supply chain management

Demand & supply forecast, inventory management, distribution optimizations, route optimization, asset tracking, supply chain risk, supplier sentiment, supplier performance, predictive maintenance

Finance
Profitability by product / channel, revenue forecasting, customer segmentation, regulatory compliance, risk mitigation

Examples:

- Retail – associate relevant customer sentiment from Hadoop with customer data in SAP Sybase IQ to enable targeted promotions and effective marketing campaigns
- Financial Services Industry – extracted data from financial reports stored in Hadoop combined with trading data in SAP HANA for rapid analysis of market trends for mission critical insights
- Telecommunications – integrate detailed machine data from Hadoop with customer data stored in SAP HANA to improve customer satisfaction

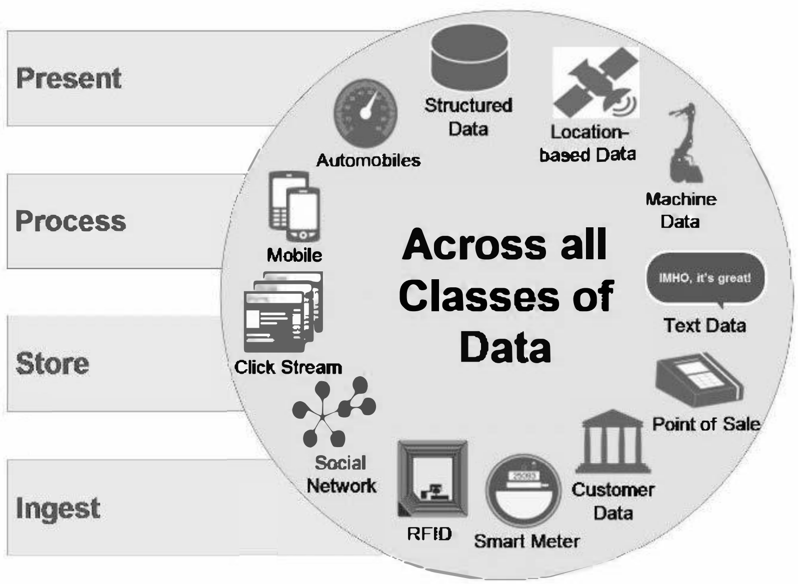


Figure 172: A Comprehensive "End-to-End" Approach is Needed

You must INGEST, STORE, PROCESS, and PRESENT all the data to get a complete and actionable picture. If you were to do a big data project on your own today you would have to do many things to put together an end to end solution. SAP solutions help you in all areas from cloud and mobile to SAP HANA that can help you achieve what you need, as well as synthesize data from other data sources like Hadoop. In this way you can optimizes storage and where the data is analyzed to best fit your needs.

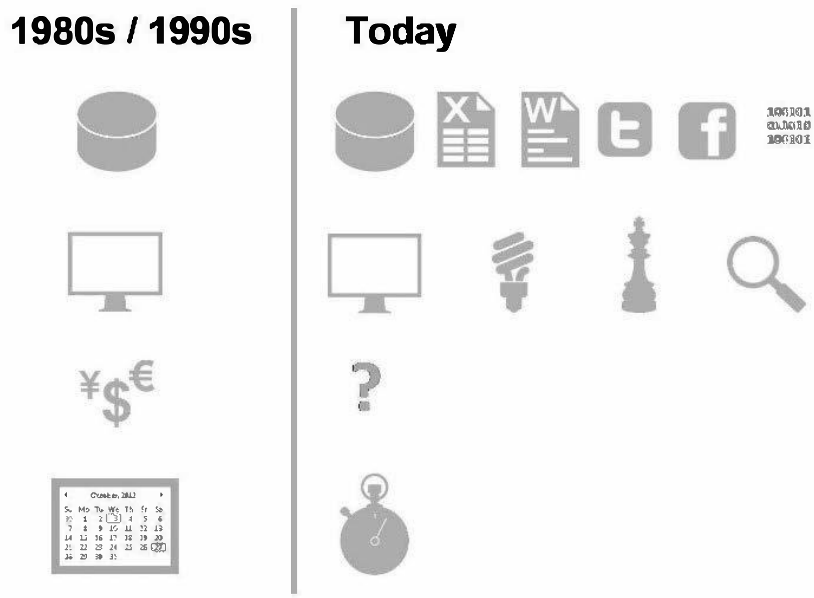


Figure 173: Traditional approaches are dead

Today you need to Transact, Analyze and Act simultaneously. Traditional approaches are dead! Today's world is very different to one that existed when the traditional relational database was first architected. Back then the database dealt with recording and storing transactional data only. What reporting happened was only for decision support. It was high value, highly structured data – not the mounds of data that may or may not have value that we face today. And it was in an era when time was measured by the calendar, not the stopwatch. Today is a completely different world. We generate vast amounts of non-transactional data – whether documents, Facebook posts, tweets, or log information coming off our phones, web servers, and other connected devices. We no longer want to just report against operational activities, but we also want to analyze, explore, predict, visualize and inspect in ways never imagined in those early days.

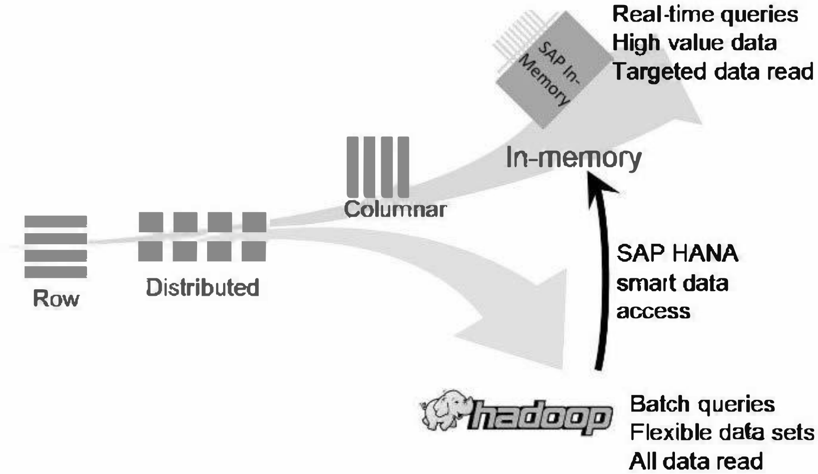


Figure 174: New storage and processing techniques required

New storage and processing techniques are now required. That's precisely why innovators have been innovating new ways to store and process data, all in an effort to get around the hard disk bottleneck. All of the approaches in essence minimize the disk bottleneck in order to improve response time. Distributed computing spreads lots of data across many disks that can all be read simultaneously.

Hadoop builds on the concept of distributed computing but opens up the platform to handle any data set, with any arbitrarily designed algorithm. Furthermore, at its core Hadoop takes a different philosophy than databases do. In particular, it assumes you want to read all (or at least most of) the data stored on your disks. So you spread the data across multiple disks. As a result the response time is now bounded by the disk with the most data divided by the transfer rate. Central to Hadoop is the idea of deploying a low-cost disk architecture. In essence, how do you read vast amounts of data off disks in a flexible, low-cost way.

The database community headed in a different direction because the type of queries they run is different. The assumption with a relational database is that only a targeted set of data needs to be accessed when responding to a query and that response times are crucial.

Like Distributed databases and Hadoop, columnar databases optimize data storage architecture in order to reduce the amount of data read off any one disk. It does this by grouping related attributes, or columns, together. The assumption is that most analytical queries only use a subset of columns, so you should only access data related to those specific columns. They also highly compress the data further reducing the number of bits read off disk.

In-memory databases take it to a whole new level.

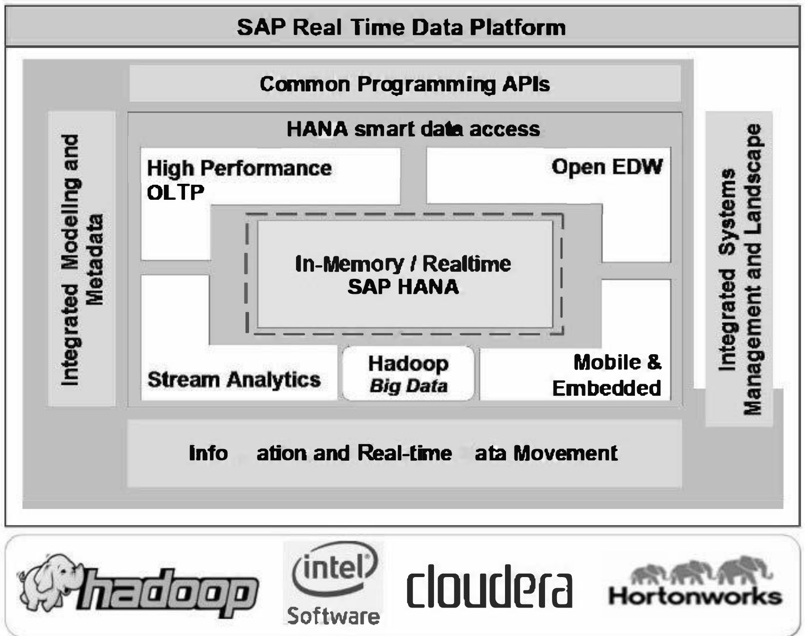


Figure 175: SAP Real Time Data Platform – A game changer

A complete Big Data solution should be end-to-end in nature. It handles everything from low-level data ingestion, storage, processing, visualization and engagement, to analytic solutions and applications. We are building an end-to-end offering that covers everything from Big Data analytic applications solving industry-specific challenges, to analytic tools, and the cover infrastructure required. It will provide for all your stakeholders as well. From your BI and analytic professionals, such as the BI Analyst, Data Scientist, IT staff – to the CIO, executive leadership, middle managers, frontline workers, and even be embedded directly into business processes. SAP believes in-memory is the future of data management, and so we're building the SAP real-time data platform, with SAP HANA at its core. And it's not just about databases. Technology like Hadoop has a critical, complementary role to play. SAP works with industry leaders such as Cloudera and Hortonworks and develops partnerships with companies like Intel around its Intel Distribution for Apache Hadoop, well as key technology and distribution partners – HP, Hitachi Data Systems, and IBM – to make it easy for customers to adopt, to give them the confidence to deploy.

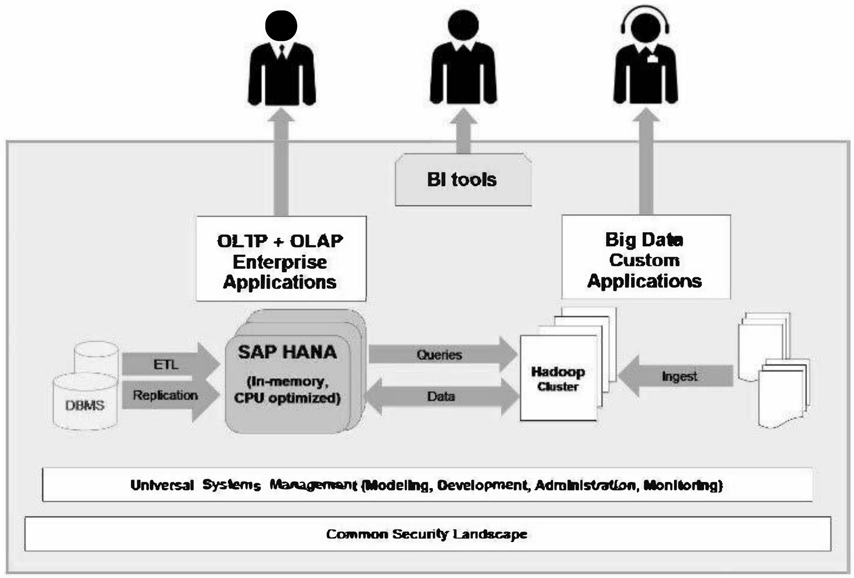


Figure 176: Realize the Value of All the Data across your Enterprise - Gain real-time insight from all the information your have

With SAP HANA smart data access can realize the value and power of in-memory on Big Data (structured or unstructured) from all your data sources to optimize business operations. SAP solution provides real-time access to and analysis of all enterprise data so they can be applied to business processes in real-time.

SAP supports Hadoop distributions from several partners including Cloudera, HortonWorks, and Intel. Our rpartnership with Intel is an extension of an existing and long relationship of joint collaboration, including in-memory innovations in SAP HANA that leverage Intel chipsets. We anticipate some of these innovations will be contributed back to the Hadoop open source community and to SAP's ecosystem of Big Data partners.

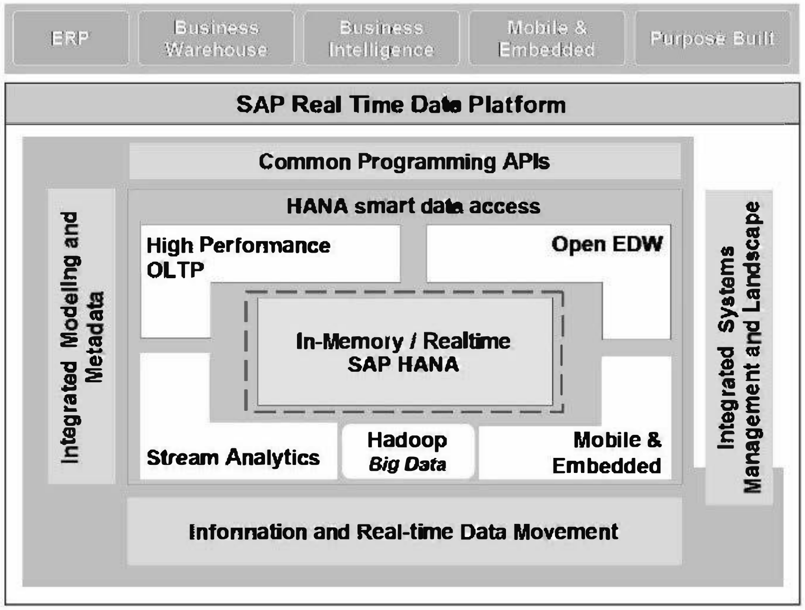


Figure 177: SAP HANA smart data access in the SAP Real-Time Data Platform (RTDP)

All data stores benefit from the superlative in-memory performance and ensuing simplification translating to simpler and iterative application design models. In an "always-on" and rapidly changing world, businesses need a real-time platform to power new experiences, with actionable insights and simplified operations to beat their competition. SAP's database and technology solutions provide powerful capabilities to help customers manage data in real-time, build and run innovative new applications on cutting-edge technologies such as in-memory, cloud and big data. SAP's real-time data platform with SAP HANA, SAP ASE, SAP IQ, SAP ESP and SAP EIM delivers comprehensive capabilities such as convergence of transactions with analytics, low latency event processing, embedded data for machines and ability to manage big data across structured and unstructured data.

SAP RTDP Foundation uses smart data access between DB's

- SAP HANA to Hadoop (HIVE)
- SAP HANA to Teradata
- SAP HANA to SAP HANA
- SAP HANA to SAP ASE
- SAP HANA to SAP IQ
- and others

Benefits

- Utilize servers that are most optimized for that type of processing, while accessing data from any other RTDP server
- All data stores benefit from the superlative in-memory performance and ensuing simplification translating to simpler and iterative application design models.
- Faster application development time.

Easily utilize enterprise wide data leveraging HANA capabilities	• Easy to setup via virtual tables, and start writing apps – using HANA features • No need to load data from source to start the project – saves cost, and is non-disruptive
High Performance	• Significant query optimizations to leverage processing capability of target sources • Move minimal data between SAP HANA and sources
Secure access to remote data	• Access remote data securely utilizing secondary credentials
Leverage Big Data processing	• Integrate output of Map-Reduce jobs in Hadoop/HIVE and access seamlessly from SAP HANA
Seamless archived data access	• Store hot data in HANA, warm data e.g. in SAP IQ and Hadoop, – get seamless access from SAP HANA • Queries in SAP HANA can integrate data from these systems

Figure 178: Value Proposition - SAP HANA smart data access

The Value Proposition for SAP HANA smart data access includes the following:

- Easily utilize enterprise wide data leveraging HANA capabilities
- High Performance
- Secure access to remote data
- Leverage Big Data processing
- Seamless archived data access

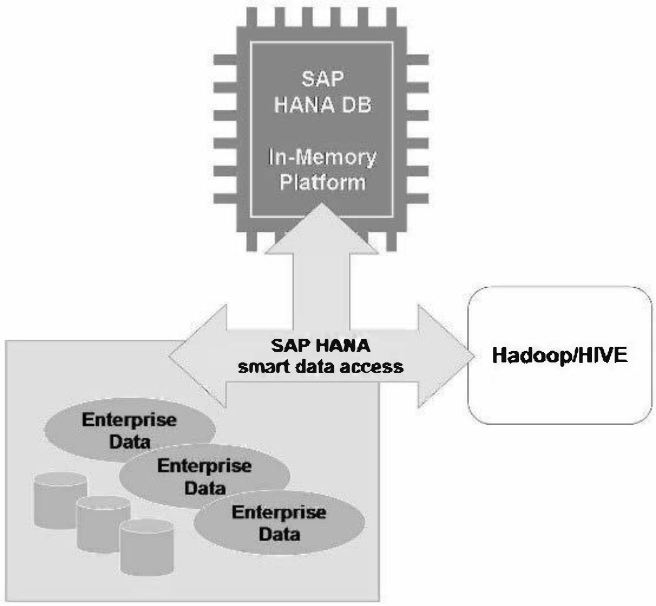


Figure 179: HANA smart data access (Data Virtualization)

HANA smart data access (data virtualization) bring the power of SAP f HANA to enterprise data:

- Enables application developers to leverage the full power of HANA – analytics libraries, predictive, text search, geospatial, etc.
- Process big data and combine with real time analytics in HANA – in a single query
- Access and synthesize enterprise data regardless of location, size, representation – without moving it
- Federation into the database layer with optimized access to remote data
- Leverage existing Corporate Data Assets and harvest Meta Data from Remote Systems

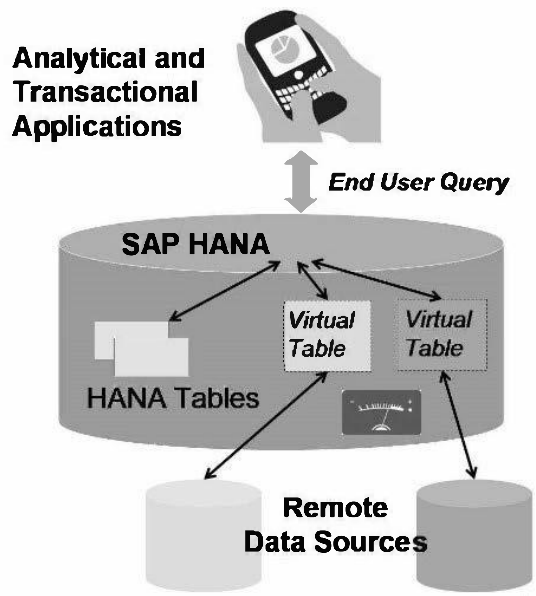


Figure 180: SAP HANA smart data access: The "glue" of the platform

SAP HANA smart data access starts becoming the "glue" of the platform.

- Provide a data platform for application developers to write new data intensive analytical applications, without regard to where the data resides, its type, quality, or size
- Platform will provide the capability to extract the required information (not data), from a variety of data sources
- Platform will leverage data processing engines (databases, Hadoop, etc) to synthesize information needed for analysis
- This will enable the rapid development of these applications, with the highest degree of performance

Some of the benefits of Smart Data Access are:

- Enables access to remote data access just like "local" table
- No special syntax to access heterogeneous data sources
- Smart query processing pushing as much processing as possible to target data source
- Smart query processing including query decomposition with predicate push-down, functional compensation
- **Functional compensation** allows customers to use the full power of the HANA
- Automatic data type translation enables remote data types to be mapped to HANA data types
- Supports data location agnostic development
- No special syntax to access heterogeneous data sources
- Provides SAP HANA to SAP HANA queries
- Non-disruptive evolution
- Support of Insert, Update and Delete (except Hadoop)
- Calculation View Support for Virtual Tables
- 'Remote Caching' for Hadoop sources
- Support for CL●B & BL●B data types
- Deliver Generic Adapter framework to extend additional Remote Sources
- hdbsdautil – utility to debug Remote Source connectivity and configuration

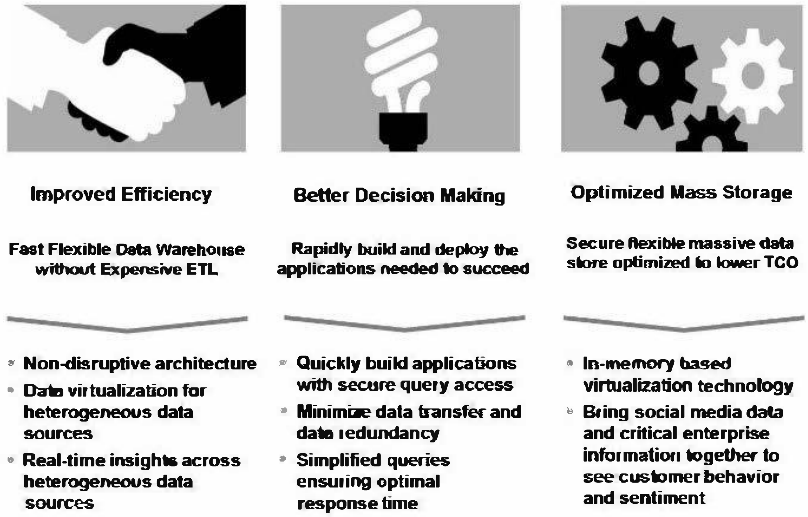


Figure 181: SAP HANA smart data access - Increase the value of your data

SAP HANA smart data access increases the value of your data. You can create a fast & flexible data warehouse without expensive ETL, massive storage, security and privacy risks. Gartner calls this a “logical data warehouse”, Forrester “data fabric” and IDC “data warehouse without ETL”. You can build big data applications with fast and secure query access to data while minimizing unnecessary data transfers and data redundancy. You can bring social media data and critical enterprise information together, giving comprehensive visibility into customer behavior and sentiment

Features:

- In-memory based virtualization technology
- Data virtualization for heterogeneous data sources
- Simplified data queries ensuring optimal response time

Benefits:

- Non-disruptive smarter data intelligent architecture
- High performance, Real-time, enterprise-wide data access
- Real-time insights across heterogeneous data sources

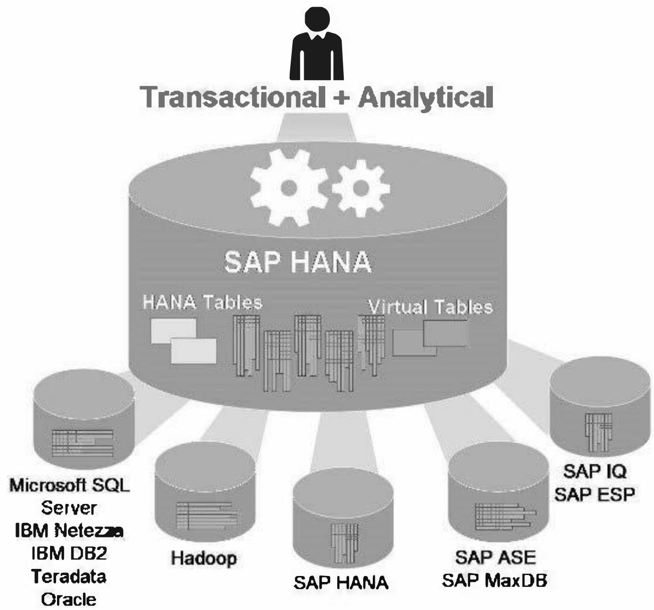


Figure 182: SAP HANA smart data access capability

Smarter Data Virtualization

SAP HANA smart data access provides non-disruptive intelligent architecture to provision enterprise-wide data from heterogeneous systems such as Hadoop, Teradata, SAP Sybase ASE and SAP Sybase IQ. As well it provides the capability for SAP HANA to SAP HANA queries which can facilitate SAP NetWeaver Business Warehouse on SAP HANA to SAP Business Suite on SAP HANA query use cases among others. SAP brings in-memory computing to data federation and it will optimize where data is stored and acted upon. Now customers can easily build unified transactional-analytical applications with secure access to data across their business networks without large data transfers. In-memory computing lets then get results from the core process in real-time, providing a single version of truth on which business can act.

Heterogeneous data sources that you can acces are:

- SAP HANA to Hadoop (Hive)
- SAP HANA to Teradata
- SAP HANA to SAP HANA
- SAP HANA to SAP ASE
- SAP HANA to SAP IQ
- Microsoft SQL Server (from SPS07)
- Oracle (from SPS07)
- IBM DB2 (from SPS08)
- IBM Netezza (from SPS08)
- SAP ESP

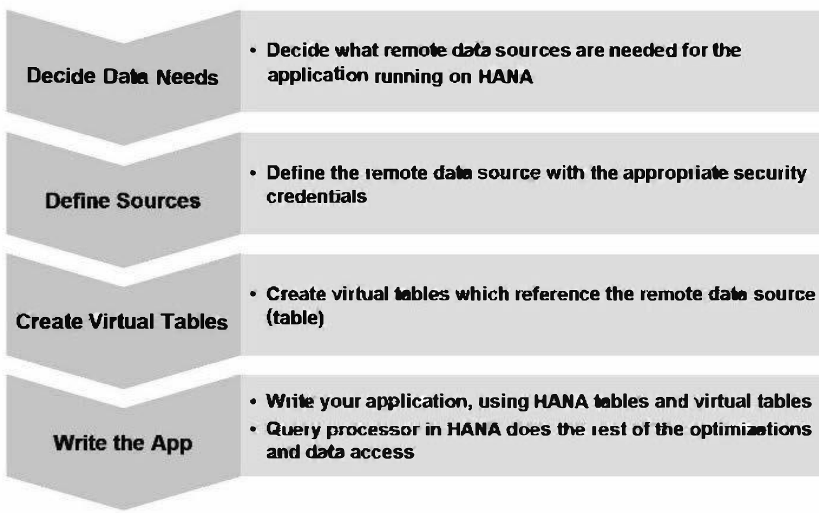


Figure 183: Simple to use

The steps to use SAP HANA smart data access are illustrated in the above diagram. Basically you first set up your remote data source, then create the virtual tables to tehse remote sources, and finally use the virtual tables in your data models.

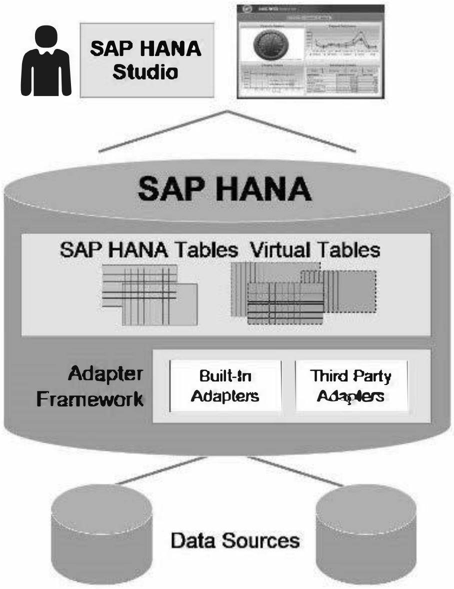


Figure 184: SAP HANA studio

You can use the SAP HANA studio when working with Smart Data Access.

- Enables users to develop applications on SAP HANA
- Create remote sources, create virtual tables, set security policies
- Access remote sources and build virtual tables, using remote table schema and data types
- Test virtual tables
- Execute queries and analyze query plan
- Monitor queries

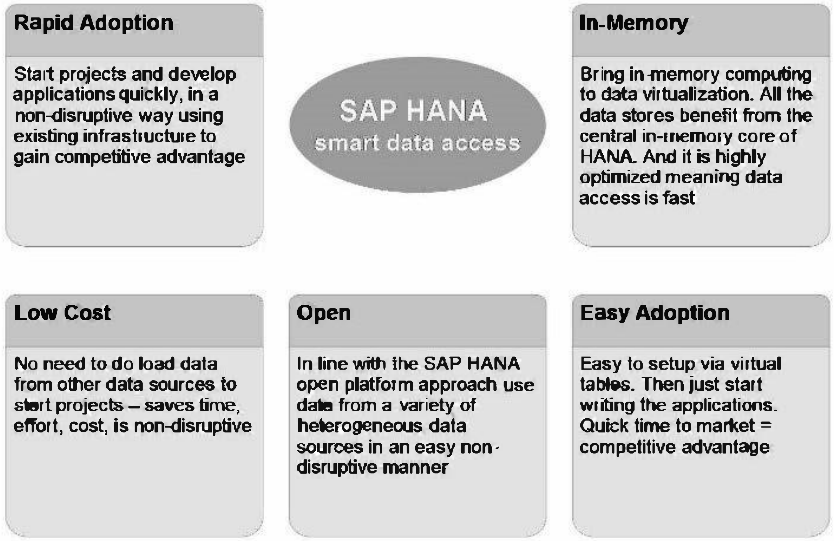


Figure 185: SAP HANA smart data access – Benefits

Using SAP HANA smart data access can lead to several clear benefits for your enterprise.

Rapid Development of data intensive analytical applications – SAP HANA smart data access brings in-memory advantages to data federation which enables enterprises to rapidly and cost effectively build data-intensive analytical applications, without moving their data around. Traditional approaches require moving all the data from its source to a central warehouse/ datamart, cleansing it and then building analytical applications, using SAP HANA smart data access, enterprises can utilize the data from its source without having to move it saving time, money effort and providing a competitive advantage.

Leveraging Big Data & Hadoop - SAP HAHA smart data access is extremely efficient, especially in situations where enterprises need to access massive amounts of data, whether they reside in databases across the enterprise, in the cloud, or on file systems, where loading this data into a central warehouse would not be a cost effective and timely option.

Extremely Fast: Data federation is extremely fast, as it allows enterprises to take advantage of their existing compute resources, and do as much processing on the remote nodes as possible, and pull in only the required information for their analysis.

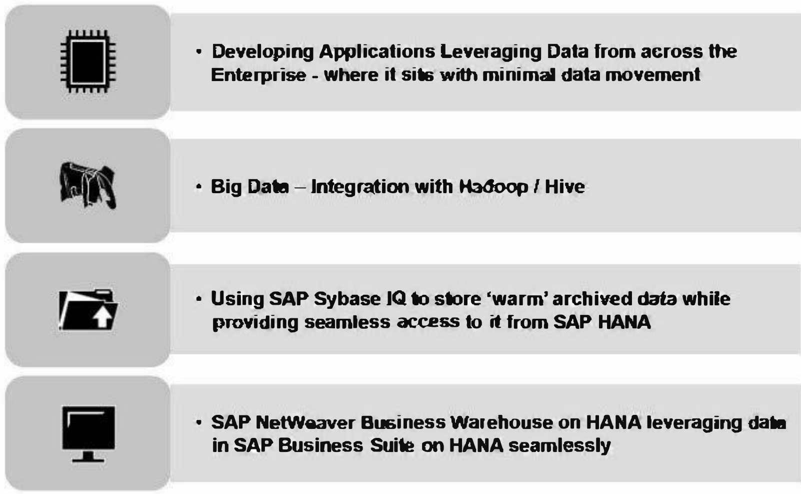


Figure 186: Use cases

SAP HANA smart data access can be used in various different ways. From integration with Hadoop to using SAP NetWeaver Business Warehouse data to data aging using SAP IQ.

Let us consider a business case.

Problem Statement
Customer is using HANA for real time offer management for their online store. One of elements of data needed for this analysis is average daily sales by product, which is kept in a Teradata point of sales system. As the point of sales data is massive, and only summarized aggregates by product are needed for the analysis, options relating to how this can be loaded into HANA are being considered.

Options Considered

- Option 1: Load the POS data into HANA and do the computation
- Option 2: Do the calculations in Teradata and load the results into HANA
- Option 3: Create virtual tables in HANA, which refer to the Teradata tables, and then use the virtual tables to compute summary statistics needed

Technical Architecture
The customer elects to go with option 3, as the summary statistics can be computed in real time, on demand and only the required data needs to be brought into HANA. To support the analysis, customer uses HANA smart data access to create virtual tables in

HANA which reference tables in Teradata. SQL statements are written for computing the summary statistics, based on the virtual tables in HANA, and this data is combined with the other information needed for creating the offers.

Benefits

- Quick application development, as not all data needs to be loaded into HANA, before the project can start
- Summary statistics based on POS data can be computed on demand, hence they are more current
- Cost effective, as ETL and other processing costs are saved

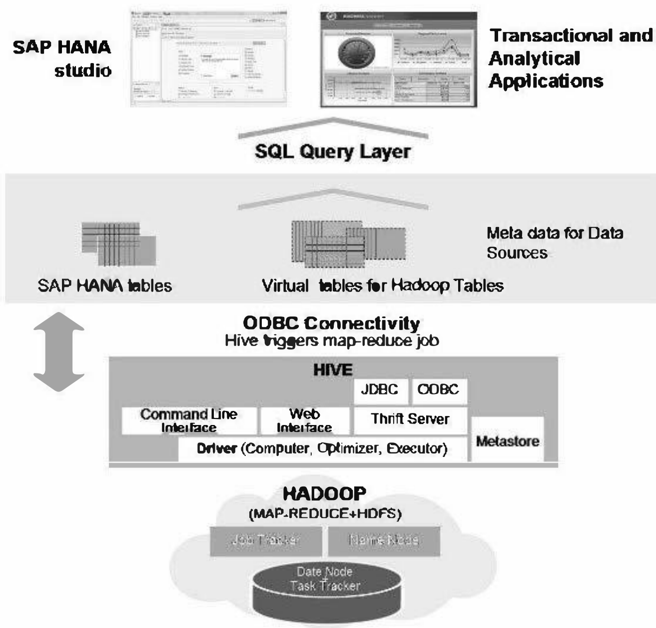


Figure 187: SAP HANA smart data access and Hadoop

Let us consider a few scenarios where SAP HANA smart data access can be used. The first is integrating to Hadoop

New Option: HANA vUDF Integration with Hadoop (SPS09)

- HANA virtual User-Defined Funtion leverages the use of native features in Hadoop (Map/Reduce, HDFS access)
- Map-Reduce Jobs allow to analyze unstructured binary data in Hadoop Cluster

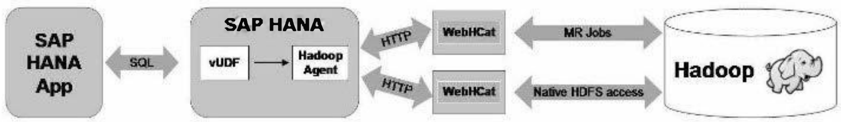


Figure 188: HANA vUDF Integration with Hadoop (SPS09)

HANA vUDF is a new type of user-defined table function and can be used wherever a normal UDF could be used (in any SQL fragment)

- The vUDF is connected to a Hadoop destination (remote source)
- The vUDF defines the mapper and reducer job classes, the input data file, job package name as well as the return (table-)parameter
- At the very first vUDF call, the associated job package is pushed and deployed into the Hadoop destination
- The map and reduce java classes can be created and stored in the HANA development environment

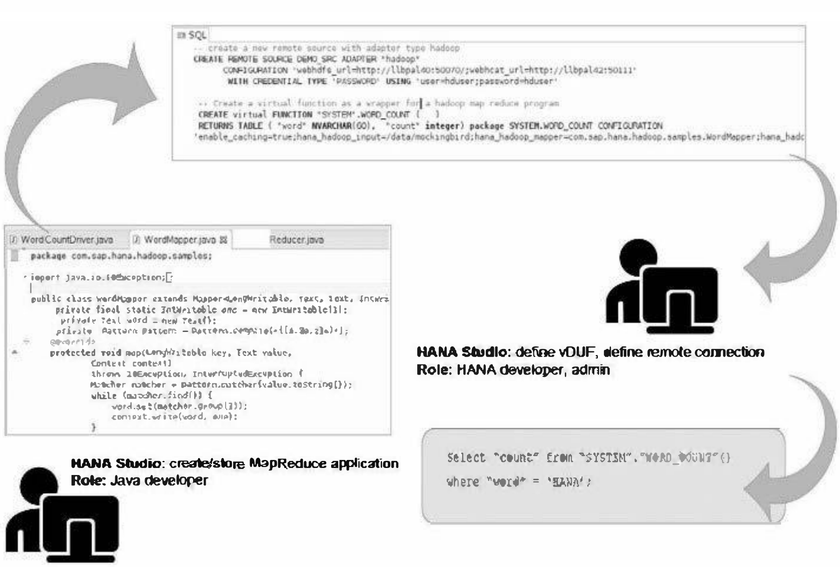


Figure 189: Programming with HANA vUDFs

Direct access to HDFS file system from SAP HANA

Develop and invoke custom Map Reduce jobs from SAP HANA

1. SAP HANA studio as the single IDE to invoke the M/R jobs
2. Leverage HANA Repository design time for Map Reduce job, remote Hadoop source and virtual functions
3. Support for remote source connectivity for IM in HANA via SAP HANA Service/Adapter Framework

Data Provisioning Support of remote result caching of virtual function execution

Support current versions of partner (Hortonworks, Cloudera etc.) Hadoop Distributions

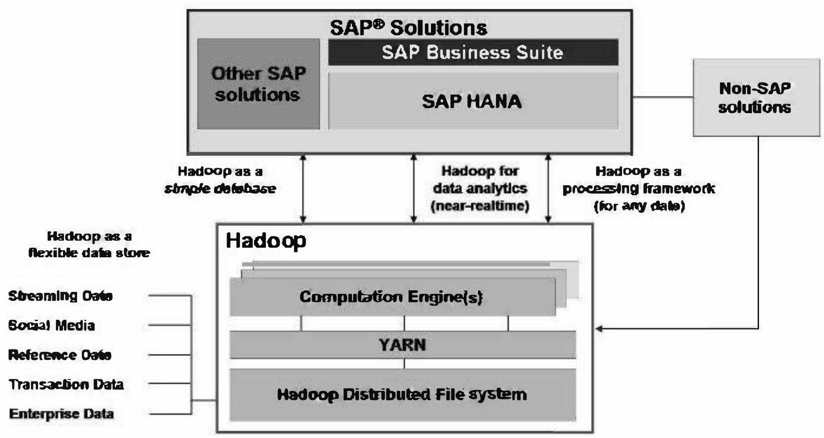


Figure 190: The potential of combining HANA & Apache's Hadoop Ecosystem

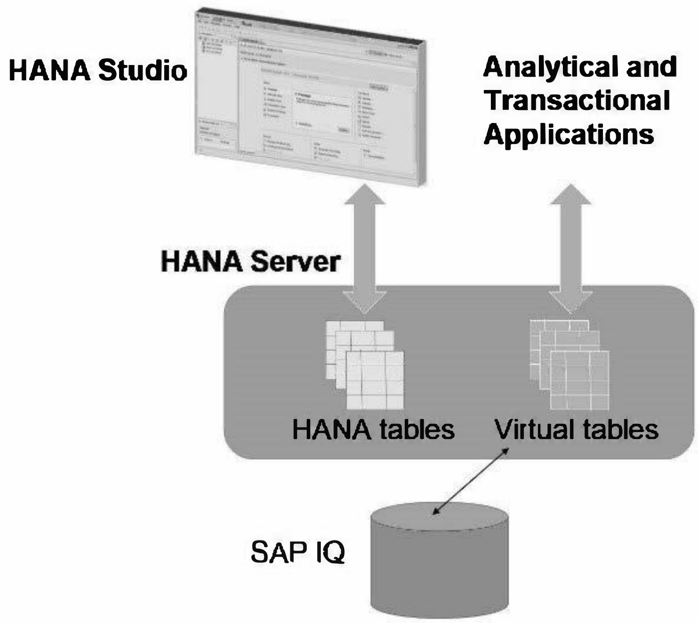


Figure 191: Use SAP IQ as a Near-line store for SAP HANA

Another scenario is using SAP HANA smart data access together with SAP IQ in a Near-line storage example. SAP IQ can be used as a cost effective archive for warm data with SAP HANA. SAP HANA contains the critical hot data and SAP IQ retains and process the archived warm data that is seamless accessible from SAP HANA.

Let us consider another business case.

Problem Statement
Customer has created a new point of sale system using HANA, and while they need 3 months of data to be online and in HANA, any data more than 3 months old is typically needed for reference purposes only.

Options Considered

- Option 1: Keep data older than 3 months in a separate system. Any time information older than 3 months is needed, customer needs to log into a different system to access this data .
- Option 2: Archive data more than 3 months old in SAP Sybase IQ. Create virtual tables in HANA, which refer to the data in IQ, and any time access to the IQ data is needed, it can be done via HANA. Additionally, archived data (older than 3 months) can be combined with current data in HANA in the same query

Technical Architecture

The customer elects to go with option 2, as they can keep archived data in IQ, while accessing it seamlessly from HANA. Using SAP Data Services, data older than 3 months is selected and moved into IQ, and deleted in HANA. Virtual tables are created in HANA which reference tables in IQ. Any time access to archived data is needed, it can be queried from HANA, via the virtual tables

Benefits

- Storage tiering can be done, with hot data in HANA, and warm in IQ
- Seamless access to archived data can be provided in HANA
- Customers can write applications in HANA which combine real time data with archived data as part of their query

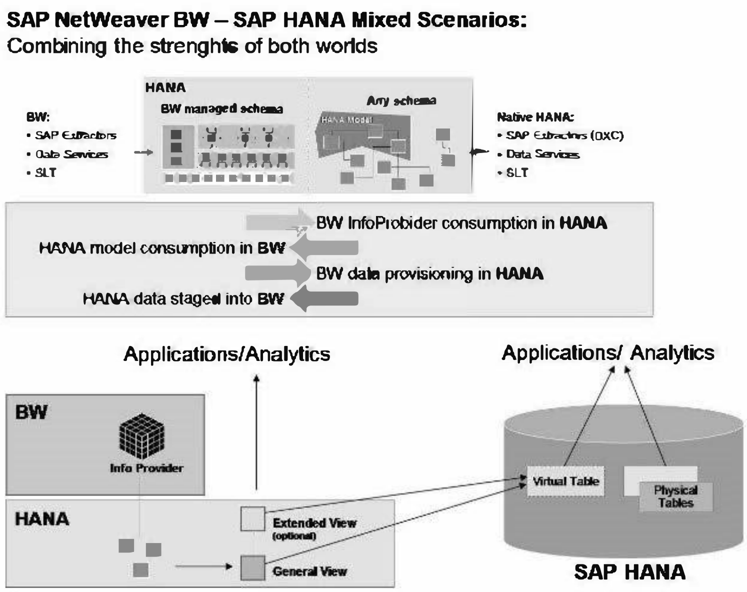


Figure 192: SAP HANA smart data access with SAP BW on SAP HANA

Another scenario where we can use SAP HANA smart data access is with SAP Business Warehouse powered by SAP HANA. You can generate SAP HANA models from the SAP BW Infocubes, and then expose these models as virtual tables in a second SAP HANA instance

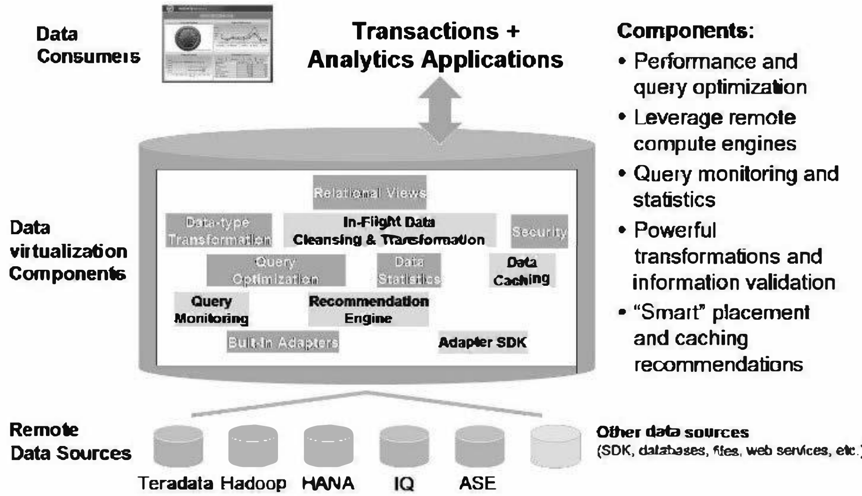


Figure 193: Components

In the above diagram the various components of SAP HANA smart data access is illustrated.

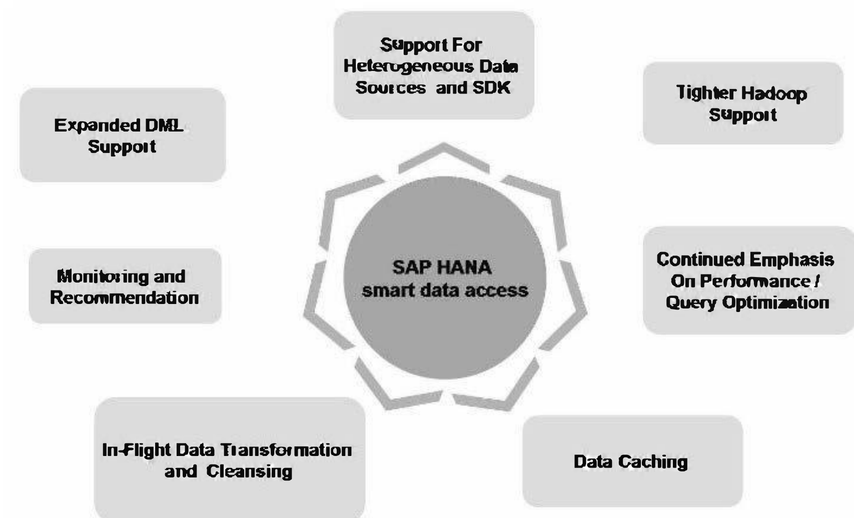


Figure 194: Focus Areas – Now and In Future

Over the last few support pack stack (SPS) releases of SAP HANA the SAP smart data access functionality has been enhanced to where it can now be used as a central "glue" for many scenarios, and a vital ingredient for our Big Data solutions.

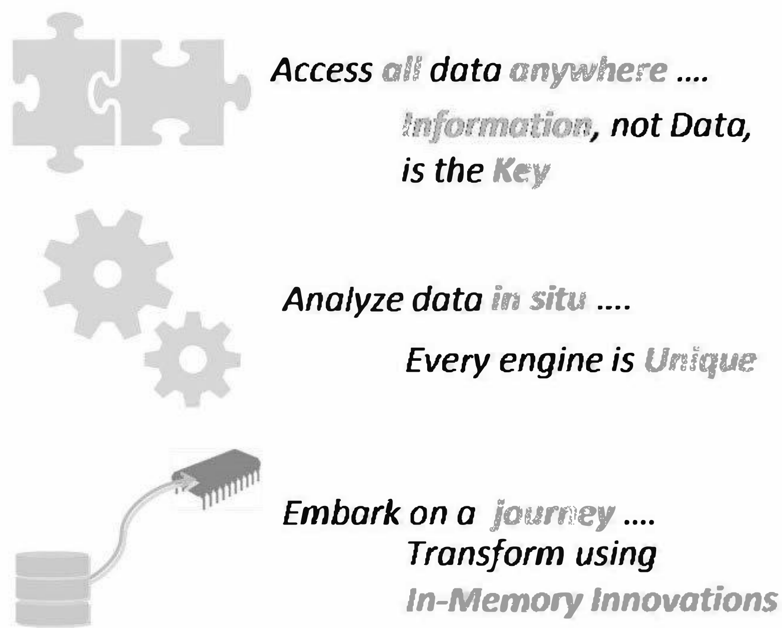


Figure 195: Platform: The whole is more powerful than its parts

Even though the various pieces can look simple, by combining them in innovative ways you can build powerful solutions to address a multitude of business challenges.

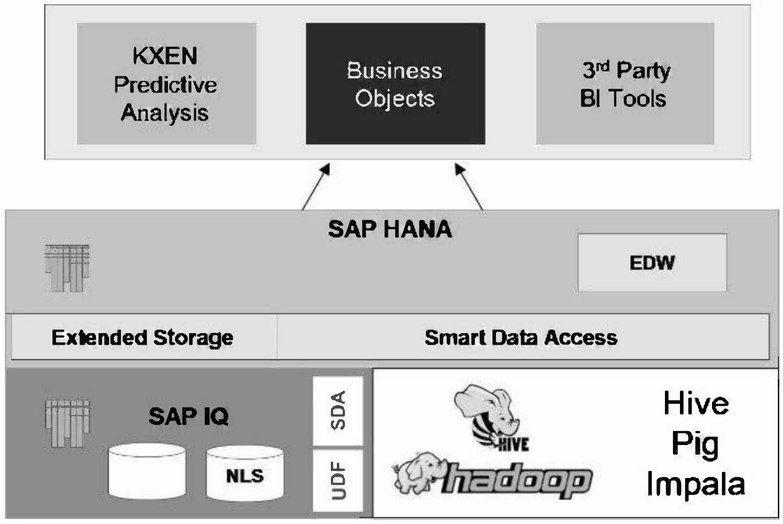


Figure 196: HANA Big Data Architecture

The SAP HANA Big Data Architecture gives us a glimpse of what is possible with these exciting new technologies.

We now have a comprehensive platform for Big Data and Analytics

- Frequency accessed data in-memory in HANA
- Dramatic reduction in time to Analysis
- Access Big Data in Hadoop directly from SAP HANA
- Leverage Hadoop from SAP IQ using SDA and UDF's
- Leverage SAP IQ from SAP HANA and SAP HANA from SAP IQ
- Store historical data in SAPIQ and Hadoop
- Leverage Column Stores query speeds
- Leverage compression for storage and processing efficiencies

Some of the expected improvements for you can achieve for Big Data are:

- Multi-Temperate Data Virtualization
- Less frequently accessed data on disk
- Process data without moving it
- Push processing into the remote source
- Reduce complexity and layers of data
- Handling Streaming Data
- Eliminate ETL with Smart Data Access and SAP BW
- Become more agile with non-materialized views

Configuring and using SDA

Configuring the driver is the only step needed to use HANA smart data access

Driver Configuration –

- Copy the ODBC driver to a HANA directory on the machine HANA is running on
- Set the environment variable LD_LIBRARY_PATH to point to the directory that the ODBC driver is in
- Optionally, edit the ODBC.INI file to add a DSN entry, or provide the remote server information in the "create remote source" SQL statement.

Figure 197: Configuration for SAP HANA smart data access

The first step for configuring Smart Data access it to ensure that the correct drivers for the relevant databases are installed on the SAP HANA server.

Configuration Validation Utility - hdbsdautil

The utility check drivers and dependency files of given data source type.

Can also connect to remote database with given credential information and run query with/without result set display.

Syntax

- *hdbsdautil -t <data source type> -S <DSN or string> [-U <username> [-P <password>]] [-Q <SQL>] [-nr] [-info]*

Examples

- *hdbsdautil -t tdodbc -S "dsn=td_dsn" -U test -P test –info*

- Try to connect to a remote Teradata database with ODBC configuration in property file (e.g. ~/.odbc.ini)
- With username and password specified
- Dump ODBC information to console

Figure 198: hdbsdautil

You can verify that the drivers have been correctly installed by using the hdbsdautil (HANA Database Smart Data Access Utility).

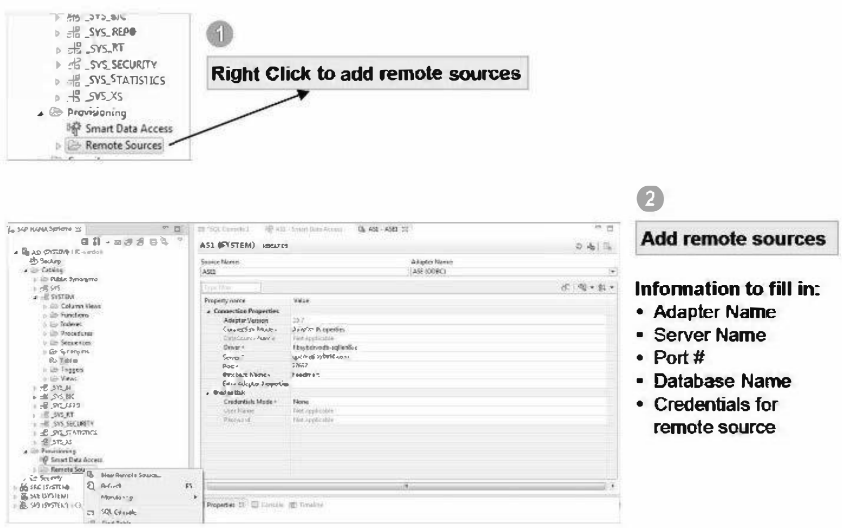


Figure 199: Creating a remote source

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Once the driver is installed you can create a link to the remote source using the SAP HANA studio. In the Provisioning folder in the Systems tab of the SAP HANA studio, you right-clikc and create a new 'remote source'. You provide a name for the remote source, select the type of database you want to connect to, and the connection details. In the exercises you see the various steps.

- **CREATE REMOTE SOURCE "HANA1" ADAPTER "hanaodbc" CONFIGURATION 'ServerNode=myhanaserver.SAP.CORP:30015;Driver=libodbcHDB.so' WITH CREDENTIAL TYPE 'PASSWORD' USING 'user=SYSTEM;password=Manager2013';**
- **CREATE REMOTE SOURCE "HIVE1" ADAPTER "hiveodbc" CONFIGURATION 'DSN=hive1' WITH CREDENTIAL TYPE 'PASSWORD' USING 'user=dftest;password=dftest';**
- **CREATE REMOTE SOURCE "IQDB1" ADAPTER "iqodbc" CONFIGURATION 'Driver=libdbodbc16_r.so;UID=DBA;PWD=sql; DBN=SAPNLSDB;ServerName=SAPNLSDB;Host=myiqserver.SAP.CORP:15050' with CREDENTIAL TYPE 'PASSWORD' USING 'user=DBA;password=sql';**
- **CREATE REMOTE SOURCE "ASEDB" ADAPTER "aseodbc" CONFIGURATION 'Server=myase.SAP.CORP;Port=17897;Driver=libsybdrvodb-sqllen8.so;Database=mydbname' WITH CREDENTIAL TYPE 'PASSWORD' USING 'user=sa;password=manager';**

Figure 200: Defining a remote data source

You can also create a remote source using SQL commands.

A remote data source is defined to HANA by associating a name to an adapter type and specifying connectivity information for that data source. Data source registration

- Data Definition Language statements (DDL) can be used
- Use ODBC DSN whenever possible!
- Credentials can be specified with this command or separately

The information in the configuration string contains the host name, the TCP/IP port number where the remote server is located, the driver files and the database name. The format and content of this clause is dependent on the adapter specified, alternatively, this information can be stored in a DSN configuration

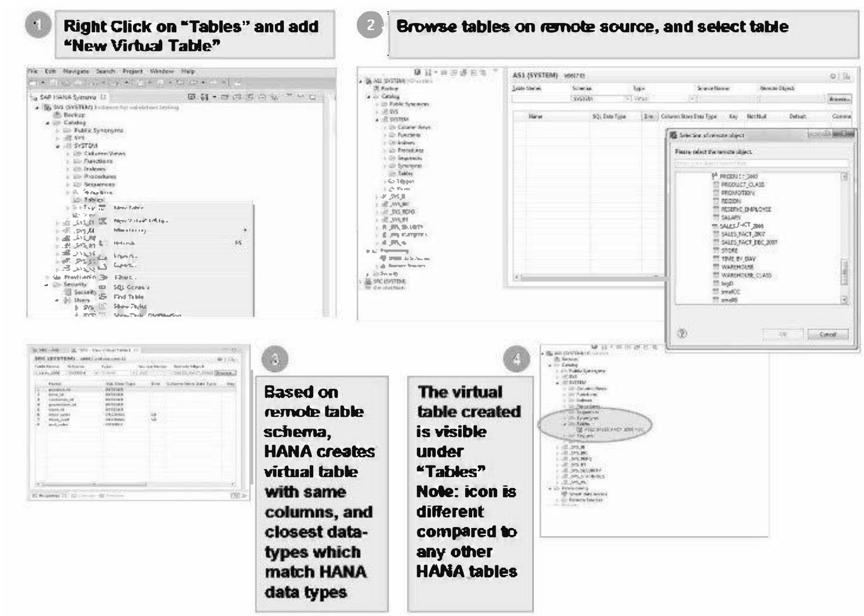


Figure 201: Steps For Creating and Using Virtual Tables

Once the remote source is created, you select the tables or views you want to work with in the remote source. You create the links to these tables or views by exposing them as virtual tables. The screenshots above show how to do this in the SAP HANA studio. The exercises also show you the detailed steps.

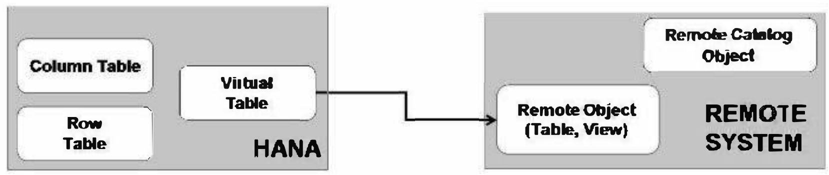
The following describe some steps for creating a virtual table:

Creating Remote Source : Creating remote source can be done using HANA Studio or using a DDL command. An example DDL command is shown below. It assumes that there is a DSN called 'hivel'. Example : CREATE REMOTE SOURCE HIVE1 ADAPTER "hiveodbc" CONFIGURATION 'DSN=hivel' WITH CREDENTIAL TYPE 'PASSWORD' USING 'user=dftest;password=dftest';

Creating Virtual Tables : Creating a virtual table for Hive needs to be done using a DDL command. An example DDL command is shown below; Example : CREATE VIRTUAL TABLE "HIVE1_PRODUCT" AT "HIVE1"."default"."default"."product";

Executing a query on virtual table: Now, that a virtual table has been created, you can simply issue a SQL query to fetch data from a HIVE table Example : SELECT * FROM HIVE1_PRODUCT;

Dropping a virtual table : For the above mentioned source, you can simply execute the command to drop a source and all of the virtual tables for that source. Example : DROP REM●TE SOURCE HIVE1 CASCADE



- Location transparency of remote data is enabled by creating a local virtual table that maps to an existing object at the remote data source site.
- The following describes the DDL statement used to create a virtual table to access any object (including tables, views) that the remote database exports as a candidate for a virtual table.
- CREATE VIRTUAL TABLE "HIVE1_PRODUCT" AT "HIVE1"."default"."default"."product";
- CREATE VIRTUAL TABLE my_schema.my_table AT remote_source.catalog.schema.object
- Remote Table datatypes, column definitions are used to create the Virtual table
- When Virtual table is created, HANA system catalog will be updated to include local column names/datatypes, remote names/datatypes, index information, etc.

Figure 202: Virtual table

As with the remote sources, you can also create virtual tables using SQL commands.

Add support for new ODBC data sources by providing connectivity configuration file

- *CREATE SOURCE DB2sample adapter "odbc" CONFIGURATION FILE 'dbs1.ini' using "Server=myDBsrv;Port=4100..."*

Sample Properties file entries

- *CAP_SUBQUERY : true*
- *CAP_SUBQUERY_GROUPBY : true*
- *CAP_SUBQUERY_SELLIST : true*
- *CAP_SUBQUERY_UPDATE : true*
- *CAP_ORDERBY : true*
- *CAP_ORDERBY_EXPRESSIONS : true*
- *CAP_ORDERBY_UNRELATED : true*
- *CAP_ORDERBY_UPDATE : true*
- *CAP_JOINS : true CAP_JOINS_OUTER : true*
- *CAP_JOINS_MIXED : true*
- *CAP_GROUPBY : true CAP_GROUPBY_ALL : true*

Figure 203: Generic Adapter Framework

If your database is not supported by native database libraries in SAP HANA smart data access yet, you could probably still access it using the Generic Adapter Framework. This uses the relevant ODBC drivers for your database, together with a connectivity configuration file as illustrated.

Provides high performance by reducing data transfer between HANA and remote data source – via query push down (push down filters, aggregates, semi-joins

Extend the HANA functionality to remote data source via functional compensation.The optimizer is aware when HANA specific features cannot be used and compensates for those

- Built-in functions
- SQL features not supported in the remote server

Functional translation to deal with syntactical differences between HANA and remote server

Figure 204: Query Processing – Overview

Although traditional optimization techniques usually apply to federated data, there are additional unique requirements:

- The optimizer must know whether certain operations can be shipped to a remote server/data source and be able to consider this in its costing.
- The optimizer must be able to generate valid plans in cases where HANA specific features cannot be used (for example RIDs, data dictionary)
- The optimizer must be extendable to include additional algorithms that may only be useful to federated data such as semi-join reduction, relocated joins to do the joins at a remote site rather than always pulling data into HANA
- A cost model that allows for network transfer and latency to be considered when choosing between plan candidates.

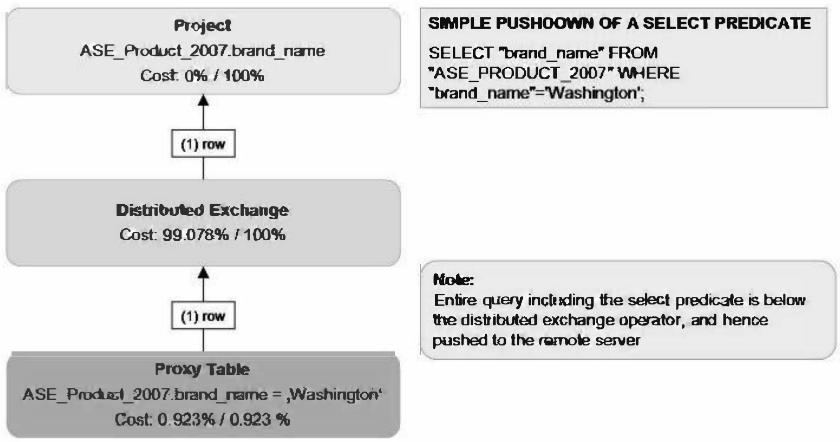


Figure 205: Simple Pushdown of a Select Predicate

The screenshot illustrates how SAP HANA pushes down the relevant parts of a SQL query to a remote source. The SQL optimizer in SAP HANA knows what each databse is capable of, and send only valid queries to these remote sources.

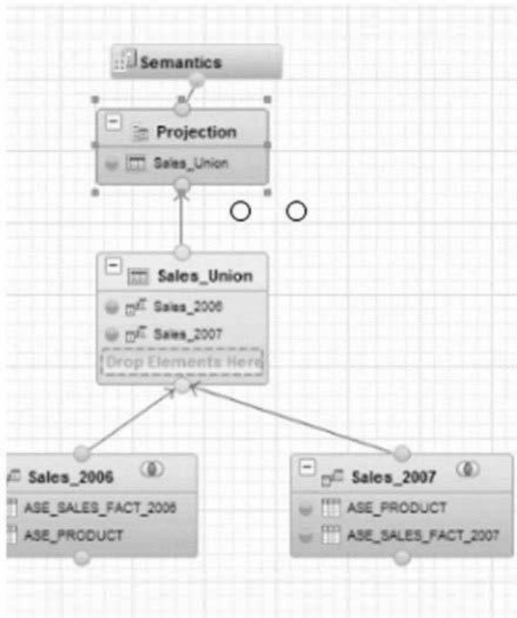


Figure 206: Calculation View support for Virtual Tables

When creating a calculation view, it is possible to add virtual tables as data sources. Virtual tables can also be referenced by the calculation scenarios. The support for adding virtual tables in the calculation view can also be done from SAP HANA Studio. Optimizations such as push down of filters is also supported in these Calculation scenarios. In the exercises, you can see all the detailed steps of how to do this.

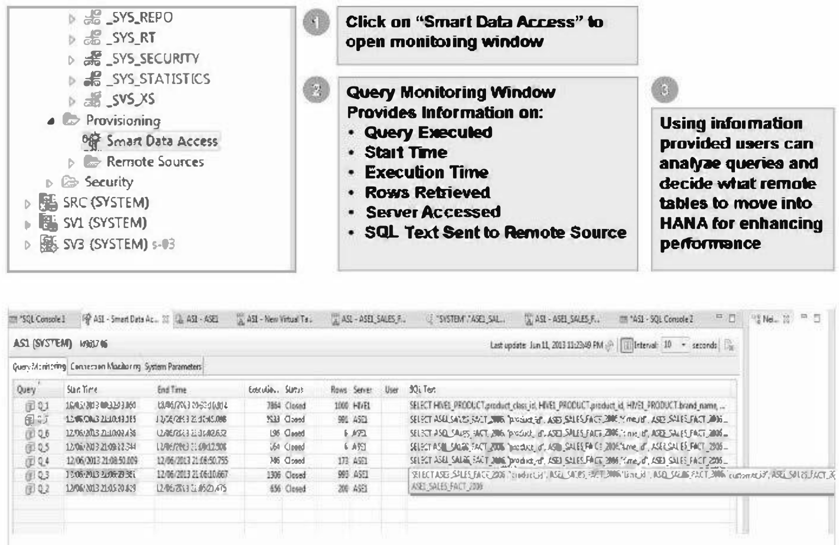


Figure 207: Query Monitoring

You can monitor remote connections and queries using the SAP HANA studio. Open the Data Provisioning folder in the Systems area in the SAP HANA studio, and select Smart Data Access.

Security

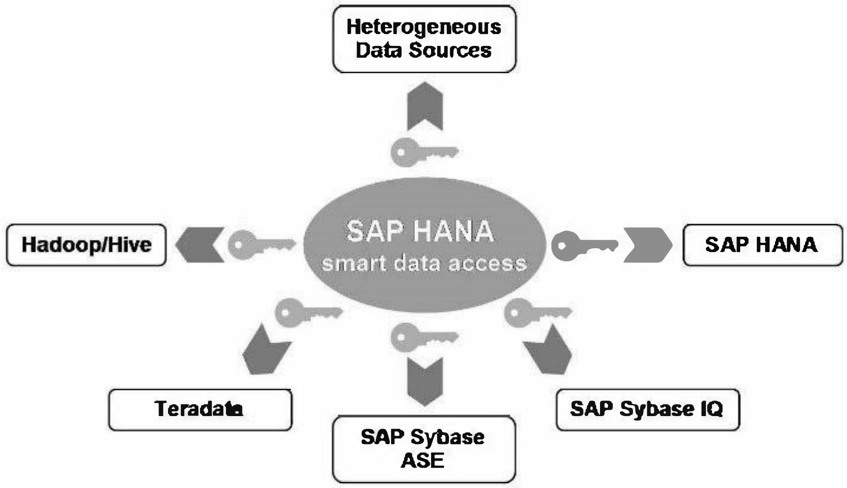


Figure 208: SAP HANA smart data access – Secure the Data

Secure Access:

- Allows access to remote servers, based on secondary credentials
- Provides the ability to support secondary credentials and technical credentials
- Technical Credentials are attached to 1 data source for all users
- Secondary credentials are for 1 user and 1 data source

Security Model

Supports the ability to create technical user credential and secondary credential

- **Technical User Credential:**
  - 1 credential attached to 1 data source for **ALL** HANA Users
- **Secondary Credential:**
  - 1 credential attached to **1 HANA User** *and* **1 Data Source**
- **Credential type: explicit user + password pair**

Category	Feature Description	Features In SP06
Security	Provide the ability to support secondary credentials and technical credentials	CREATE SOURCE <Data Source name> ADAPTER "<predefined adapter type name>" USING '<connector specific properties like adapterProperties:DSN=mydsn>' [WITH TECHNICAL USER CREDENTIAL <Credential>]
	Technical Credentials are attached to 1 data source for all users	CREATE SECONDARY CREDENTIAL [FOR <Hana User name>] ON <Data Source name> WITH <Credential>;
	Secondary credentials are for 1 user and 1 data source	DROP SECONDARY CREDENTIAL [FOR <Hana User name>] ON <Data Source name>
		DROP SOURCE <Data Source name> [<drop options>]

Figure 209: Security Model

Exercise 3: Using Smart Data Access with an external SAP ASE Database

Exercise Objectives

After completing this exercise, you will be able to:

- Create a virtual table using Smart Data Access
- Create a calculation view using a virtual table

Business Example

You are asked to create a sales report from data stored in SAP ASE and SAP HANA. You do not wish to increase the memory consumption of the SAP HANA system with data for a sales report that is not run very often.

Task:

Create a report that shows the total amount made per book title sold. The book information for the report is stored in an SAP ASE database. You should combine this with transactional data already stored in the SAP HANA system.

1. Create a virtual table in the SAP HANA Studio with book information from the ASE database. The Smart Data Access database connection to ASE is already created. Connect to the **titles** table in ASE using a virtual table

2. Create a calculation view where you join the data from your **titles** virtual tables to the **salesdetail** table. Show only the book title and the total sales amount for each book title. Show the results in the SAP HANA studio using a treemap graphic.

Solution 3: Using Smart Data Access with an external SAP ASE Database

Task:

Create a report that shows the total amount made per book title sold. The book information for the report is stored in an SAP ASE database. You should combine this with transactional data already stored in the SAP HANA system.

1. Create a virtual table in the SAP HANA Studio with book information from the ASE database. The Smart Data Access database connection to ASE is already created. Connect to the **titles** table in ASE using a virtual table

   a) Open the SAP HANA studio. In the Systems tab, open the Provisioning folder and the Remote Data Sources. You will see a connection to an SAP ASE database that has already been created for you by the system administrator. It connects to the pubs2 database.

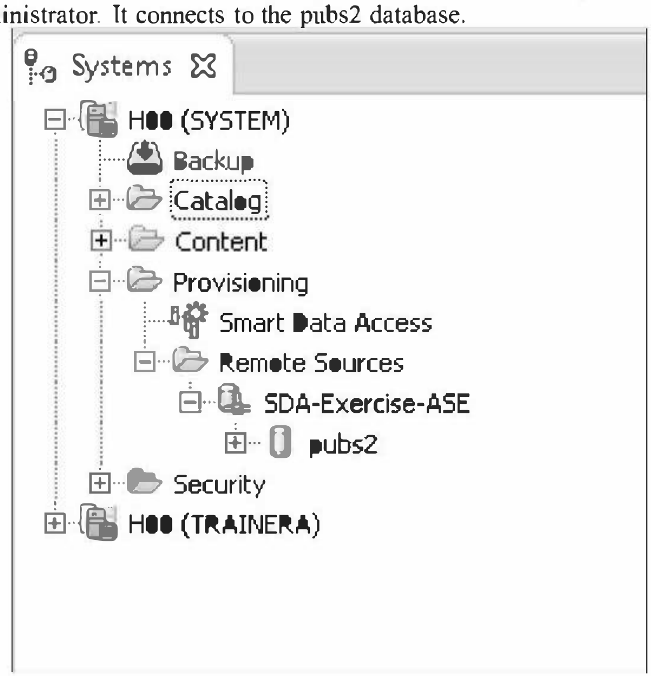


Figure 210: Smart Data Access connection to SAP ASE

   b) Double-click on the *Sample* (or *SDA-Exercise-ASE*) connection to see the connection details.

Continued on next page

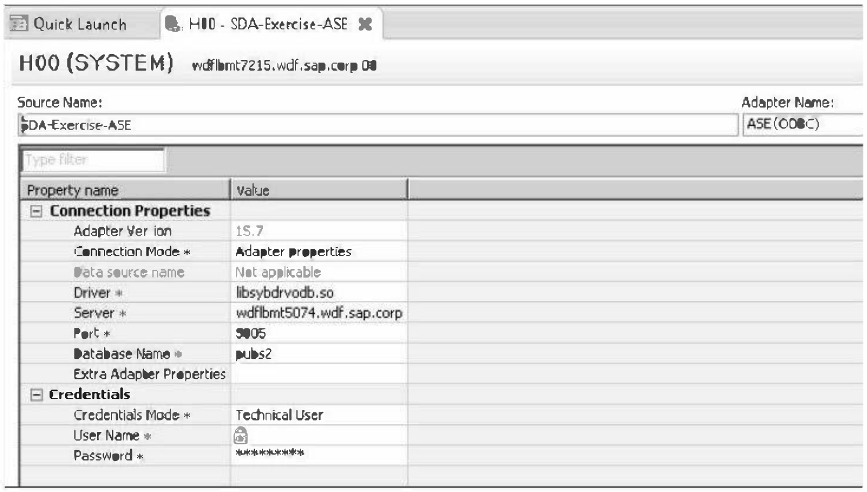


Figure 211: Details of the Smart Data Access connection to the SAP ASE database

c) Open the tree structure under the *Sample* (or *SDA-Exercise-ASE*) connection to reveal all the tables in the pubs2 database. Find the table called **titles** in the pubs2 database. Right-click on this table and select the **Add as Virtual Table** option.

Continued on next page

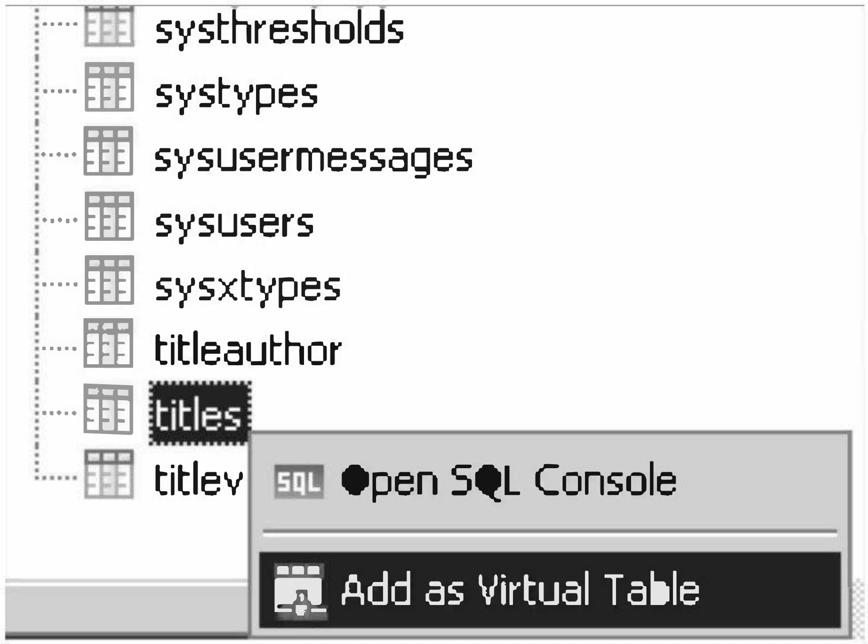


Figure 212: Add the titles tables as a virtual table

d) Name this virtual table **SDA-Exercise-ASE_titles##**, where ## is your student number (a number between 01 and 10). You can use the **SYSTEM** schema for this exercise.

Continued on next page

SDA-Exercise-ASE_titles##

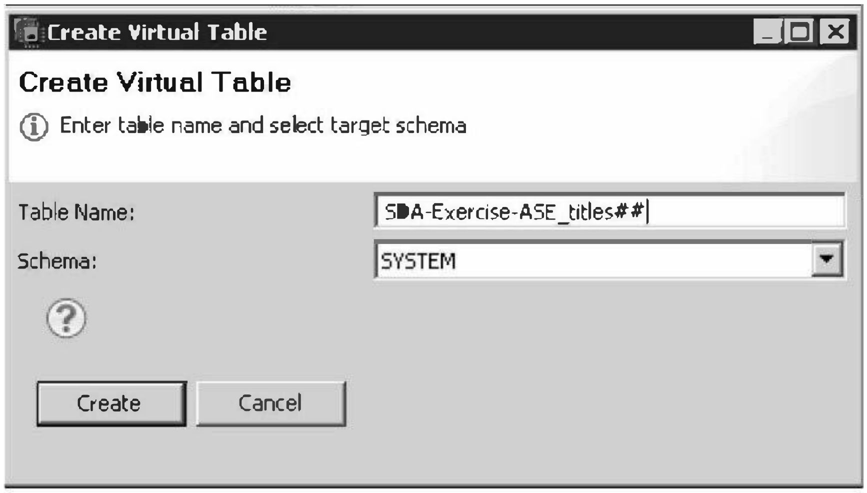


Figure 213: Name the virtual table

e) The virtual table was now successfully created.

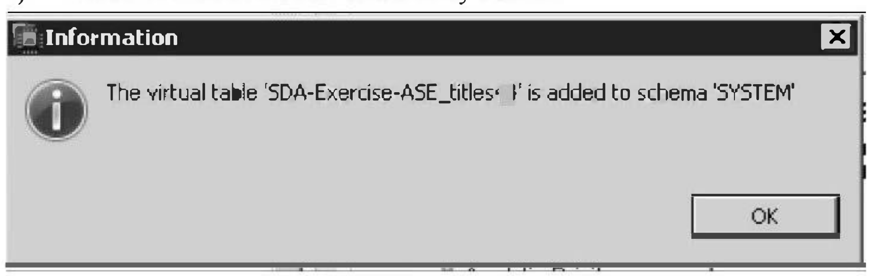


Figure 214: The virtual table was created

f) You can now view the data in the virtual table you created. Open the Catalog folder in the H00 system. Find the SYSTEM schema, and in this schema find your SDA-Exercise-ASE_titles## table. (## is your student number.) You will notice the icon for this table indicates it is a virtual table. Right-click the table name, and select the Open Data Preview menu option.

Continued on next page

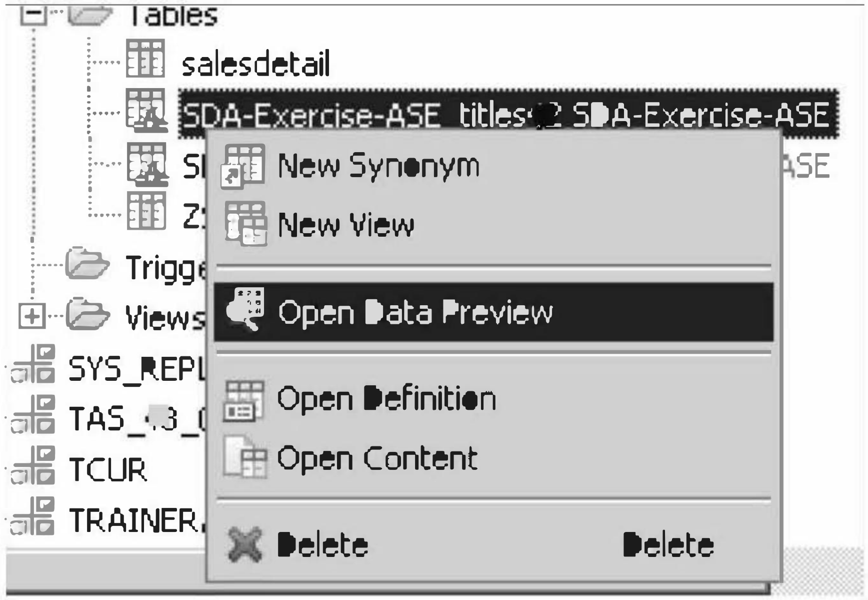
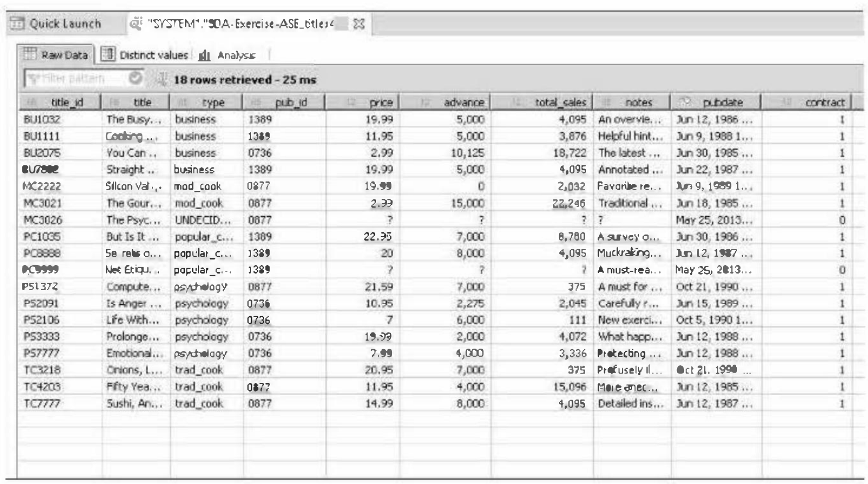


Figure 215: Preview the virtual table

g) You can now see the names and information of various books.

You have successfully created a virtual table.



title_id	title	type	pub_id	price	advance	total_sales	notes	pubdate	contract
BU1032	The Busy…	business	1389	19.99	5,000	4,095	An overvie…	Jun 12, 1986 …	1
BU1111	Cooking …	business	1389	11.95	5,000	3,876	Helpful hint…	Jun 9, 1988 1…	1
BU2075	You Can …	business	0736	2.99	10,125	18,722	The latest …	Jun 30, 1985 …	1
BU7832	Straight …	business	1389	19.99	5,000	4,095	Annotated …	Jun 22, 1987 …	1
MC2222	Silicon Val…	mod_cook	0877	19.99	0	2,032	Favorite re…	Jun 9, 1989 1…	1
MC3021	The Gour…	mod_cook	0877	2.99	15,000	22,246	Traditional …	Jun 18, 1985 …	1
MC3026	The Psyc…	UNDECID…	0877	?	?	?	?	May 25, 2013…	0
PC1035	But Is It …	popular_c…	1389	22.95	7,000	8,780	A survey o…	Jun 30, 1986 …	1
PC8888	Secrets o…	popular_c…	1389	20	8,000	4,095	Muckraking…	Jun 12, 1987 …	1
PC9999	Net Etiqu…	popular_c…	1389	?	?	?	A must-rea…	May 25, 2013…	0
PS1372	Compute…	psychology	0877	21.59	7,000	375	A must for …	Oct 21, 1990 …	1
PS2091	Is Anger …	psychology	0736	10.95	2,275	2,045	Carefully r…	Jun 15, 1989 …	1
PS2106	Life With…	psychology	0736	7	6,000	111	New exerci…	Oct 5, 1990 1…	1
PS3333	Prolonge…	psychology	0736	19.99	2,000	4,072	What happ…	Jun 12, 1988 …	1
PS7777	Emotional…	psychology	0736	7.99	4,000	3,336	Protecting …	Jun 12, 1988 …	1
TC3218	Onions, L…	trad_cook	0877	20.95	7,000	375	Profusely il…	Oct 21, 1990 …	1
TC4203	Fifty Yea…	trad_cook	0877	11.95	4,000	15,096	More anec…	Jun 12, 1985 …	1
TC7777	Sushi, An…	trad_cook	0877	14.99	8,000	4,095	Detailed ins…	Jun 12, 1987 …	1

Figure 216: Book names displayed from the SAP ASE database

Continued on next page

2. Create a calculation view where you join the data from your **titles** virtual tables to the **salesdetail** table. Show only the book title and the total sales amount for each book title. Show the results in the SAP HANA studio using a treemap graphic.

   a) Open the Content folder in the H00 system (in the Systems tab). Right-click in the **Content** folder, and elect **New**, then **Package** to create a new package.

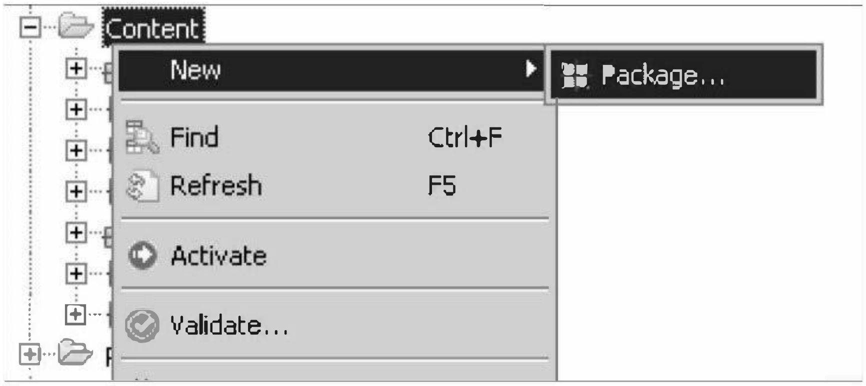


Figure 217: Create a new package

   b) Name the package SDA## (where ## is your student number). You can leave the delivery unit field blank. Then click OK.

Continued on next page

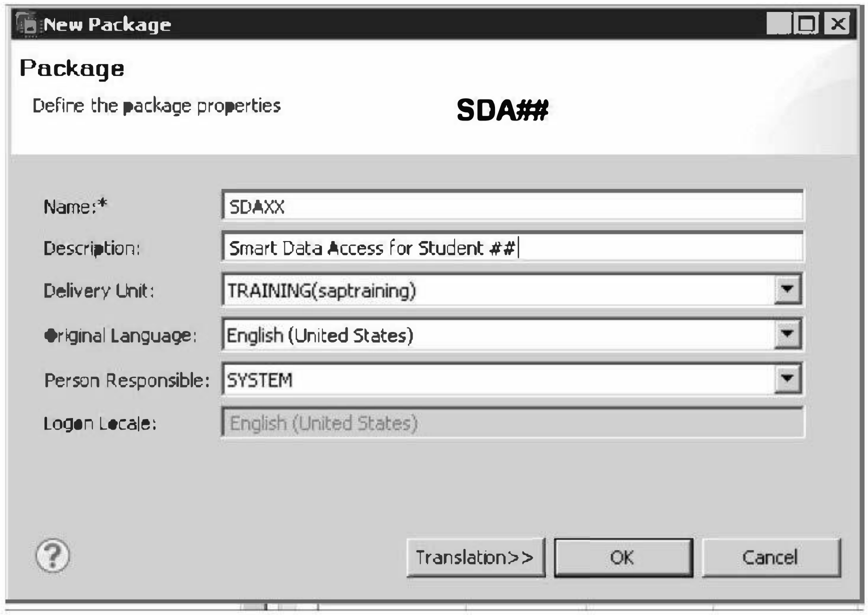


Figure 218: Create a package named SDA## (where ## is your student number)

c) Create a new calculation view by right-click on your new package name, then selecting **New** and **Calculation View** from the menu.

Continued on next page

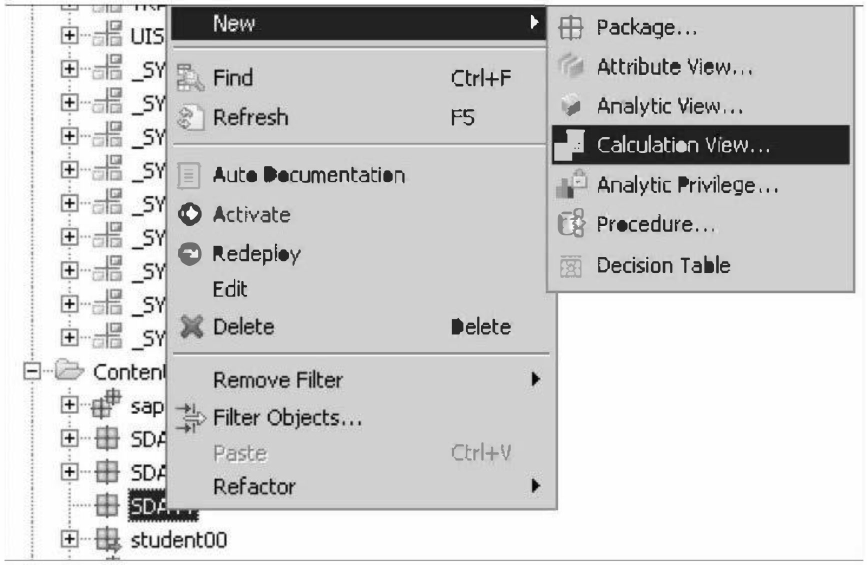


Figure 219: Create a new calculation view

d) Name your calculation view CV_SDA_##. (CV for Calculation View, SDA for Smart Data Access, and ## is your student number.) Click on Finish to create the calculation view.

Continued on next page

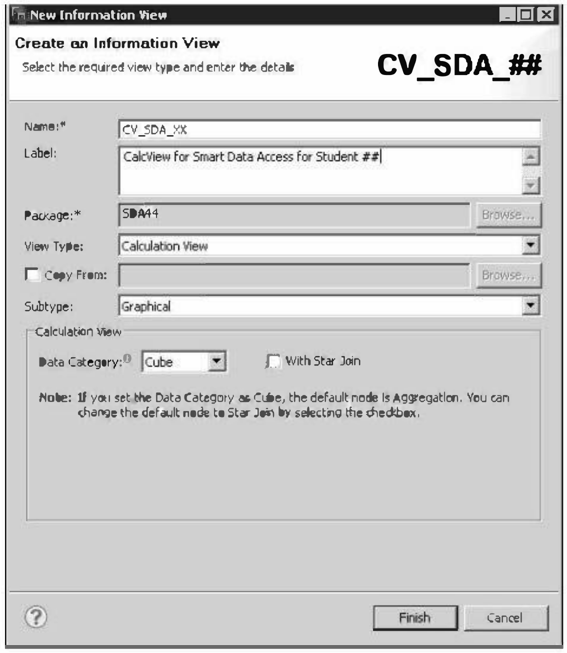


Figure 220: Create a new calculation view named CV_SDA_## (where ## is your student number)

c) You want to join two tables, so add a join node by click on the Join picture (of two overlapping circles) and then on the blank space under the aggregation node.

Continued on next page

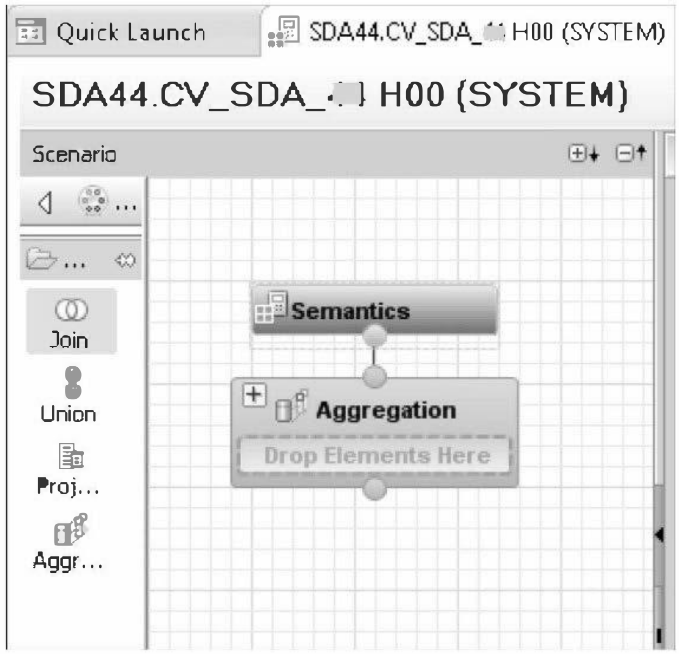


Figure 221: Add a join node below the aggregation node

f) Link the join node to the aggregation node by drawing a line from the connector point (middle top) of the join node to the connector point (middle bottom) of the aggregation node.

Continued on next page

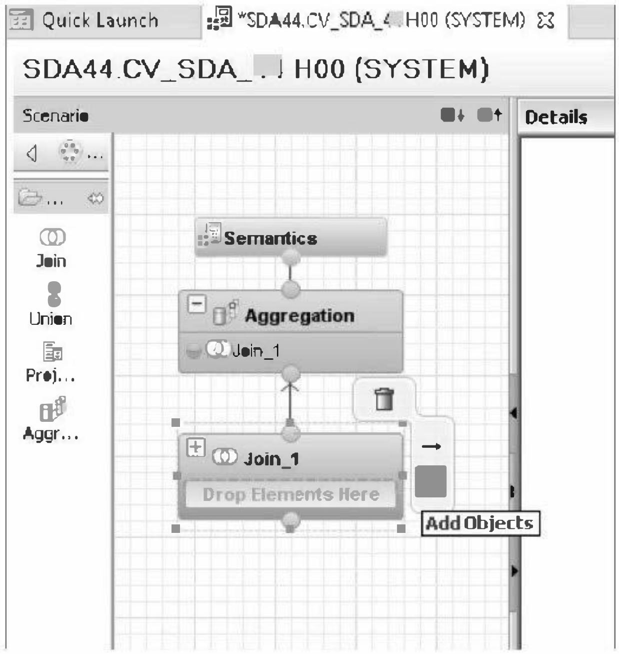


Figure 222: Link the join node to the aggregation node

g) Select the join node. Add the virtual table you created and the salesdetail tables to the join node.

You can add tables to the join node in twoways:

1. Open the Catalog area in the Systems tab area, and open the SYSTEM schema. Drag the 2 tables from the SYSTEM schema to the join node.
2. Hover over the join node. A + icon will appear on the right of the join node (see the previous screenshot). Click on this + icon. A search box will appear where you can find your titles virtual table.

Continued on next page

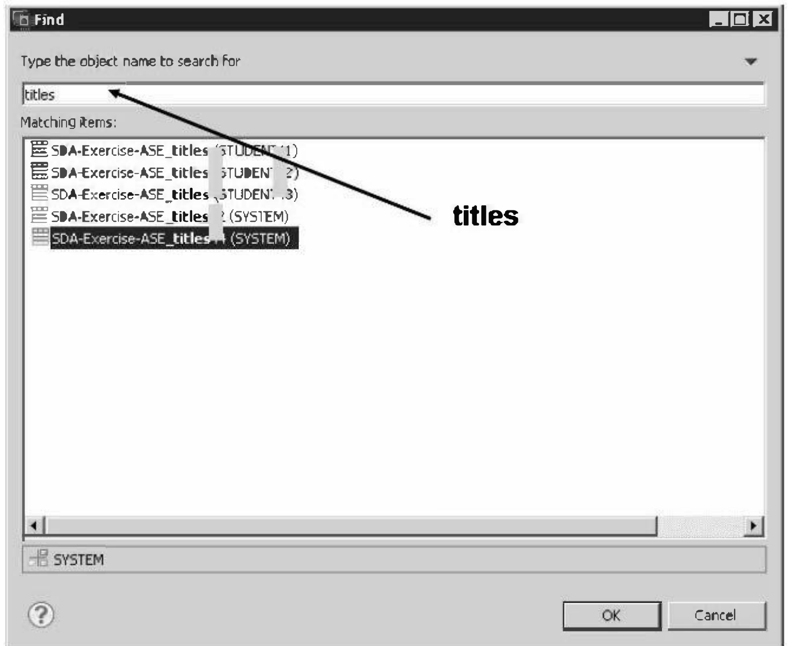


Figure 223: Find and add your titles virtual table

h) You can use either of the two methods. (Illustrated by the screenshots above and below.)

Add your virtual table and the salesdetail tables to the join node.

Continued on next page

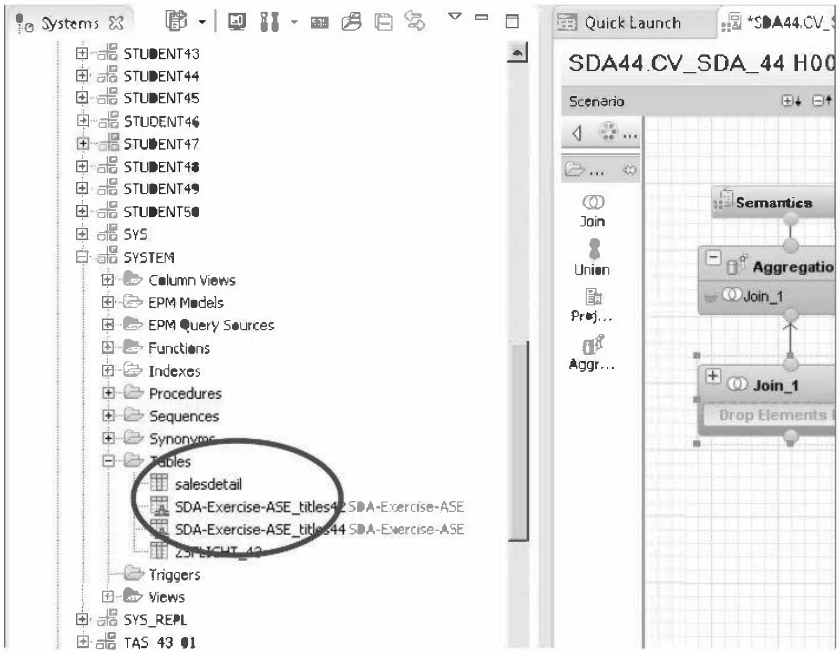


Figure 224: Also add the salesdetail table to the join node

i) Now join the two tables (in the middle area) by dragging the title_id field from the titles table, and dropping it on the title_id field in the salesdetail field.

Continued on next page

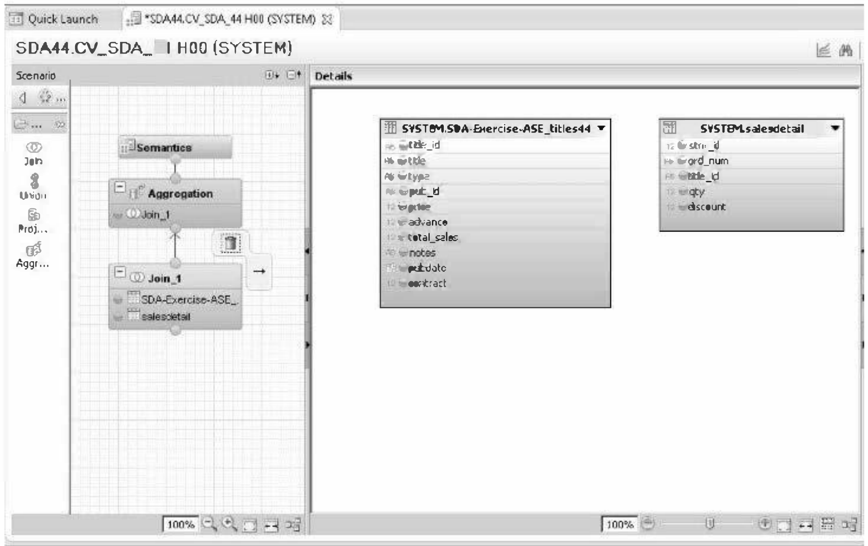
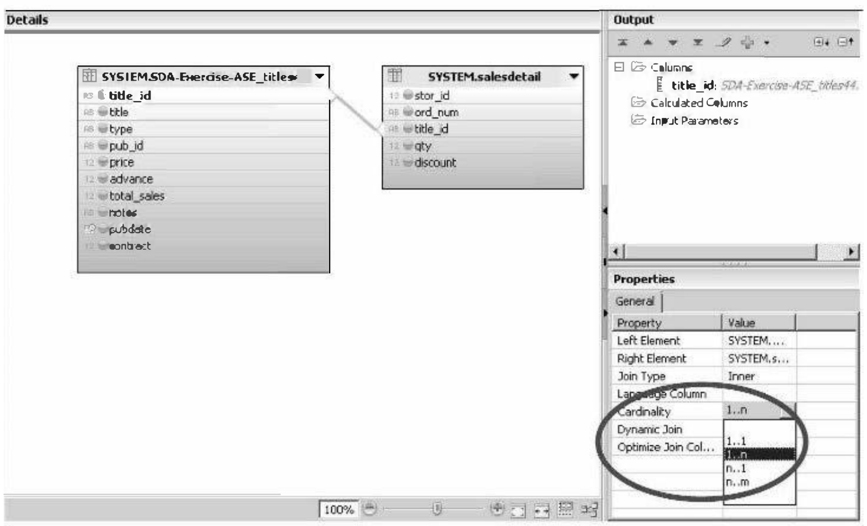


Figure 225: Join the 2 tables on the title_id fields

j) Specify the cardinality as 1..n

The cardinality is 1..n because you have 1 book title, and many transactions (sales) for each book title.A



Continued on next page

Figure 226: Check the cardinality of the join

k) Add the following fields to the output. Simply click on the dot next to each field name, or right-click and select Add to output.

- title_id (was already selected when you created the join)
- title (from the titles table)
- price (from the titles table)
- qty (from the salesdetail table)
- discount (from the salesdetail table)

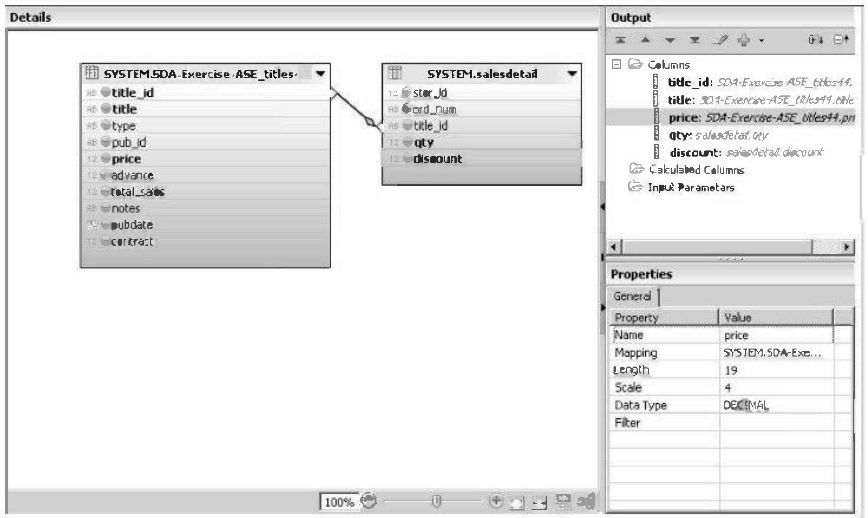


Figure 227: Add fields to the output

l) You now need to calculate the sales amount for each transaction per book title.

On the right side under Output, right-click on the Calculated Columns, and select New.

Continued on next page

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

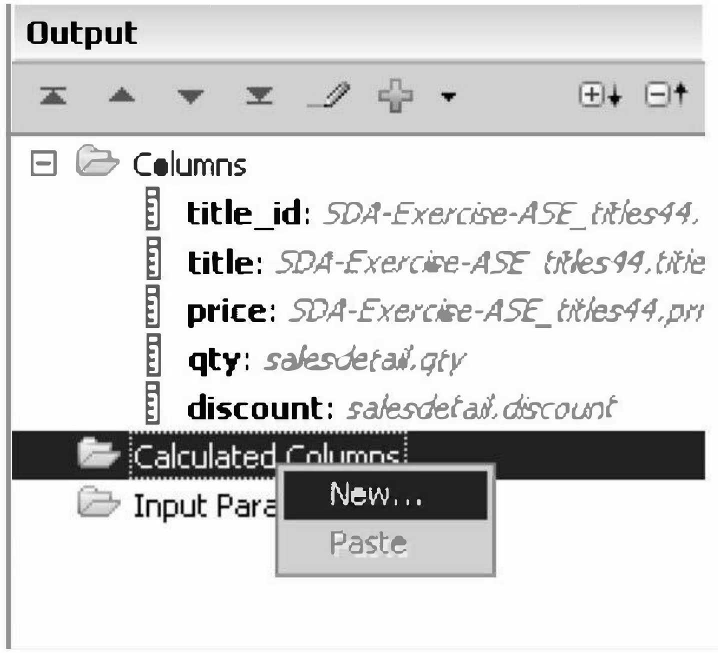


Figure 228: Create a new calculated field

m) Call the calculated field **Amount**

Specify the data type as **Decimal** with **15, 2**

Specify the formula as: "price" * "qty" * (100-"discount") / 100 to calculate the actual sales amount with discounts included.

Then click OK

Continued on next page

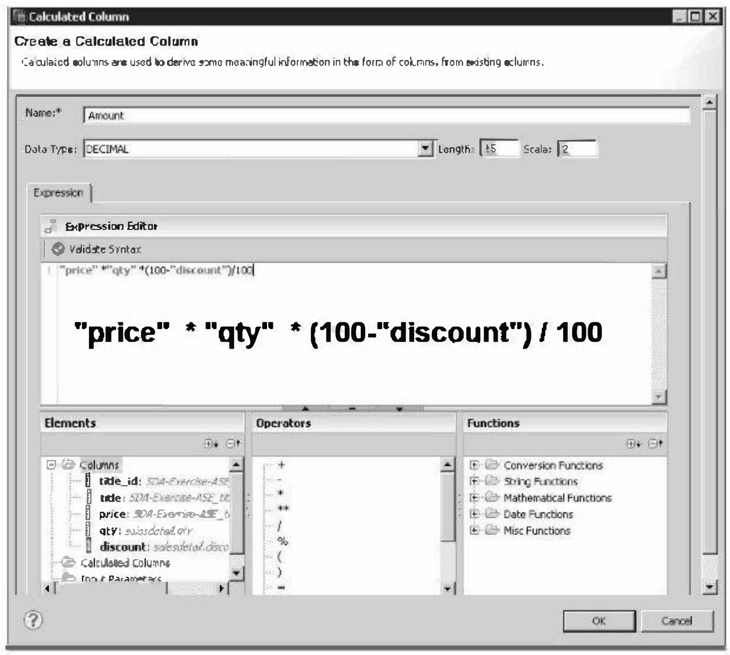


Figure 229: Enter the formula for calculating the sales amount, applying the discount

n) Now select the aggregation node by click on the block.

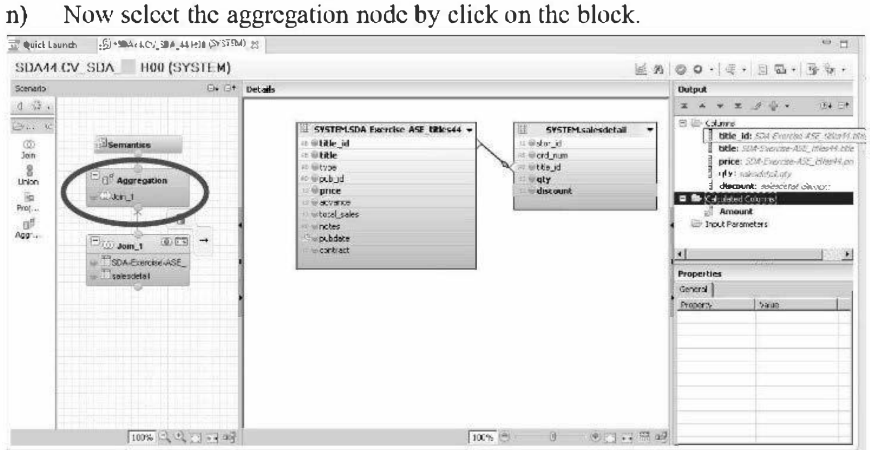


Figure 230: Go to the aggregation node

Continued on next page

o) You only need the book name and sales amount in the report. So only add those two fields to the output.

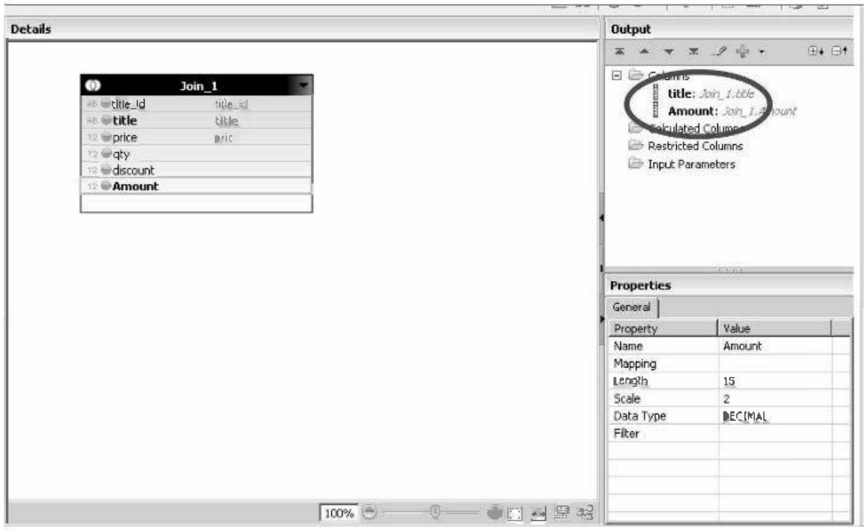


Figure 231: Add the book title and the sales amount to the output

p) Click on the Semantic block to select it.

You now have to specify each output field as either an attribute or as a measure. You can easily do this by click on the Auto assign button as indicated.

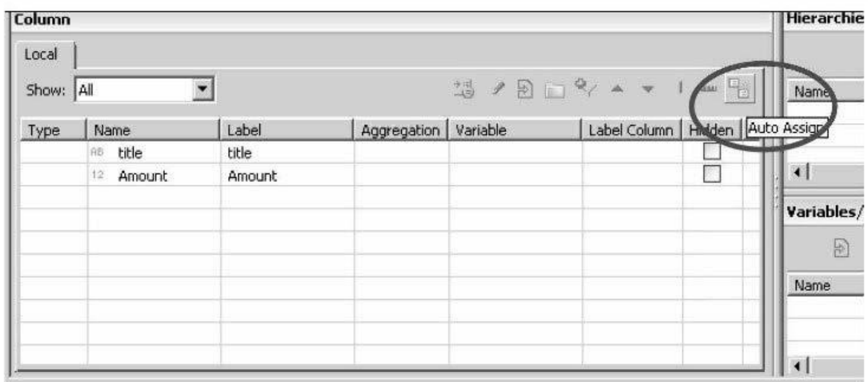


Figure 232: Go to the semantic node, Auto assign

Continued on next page

q) The fields are now automatically assigned type as attribute or measure. You can see this in the left column by looking at the icons.

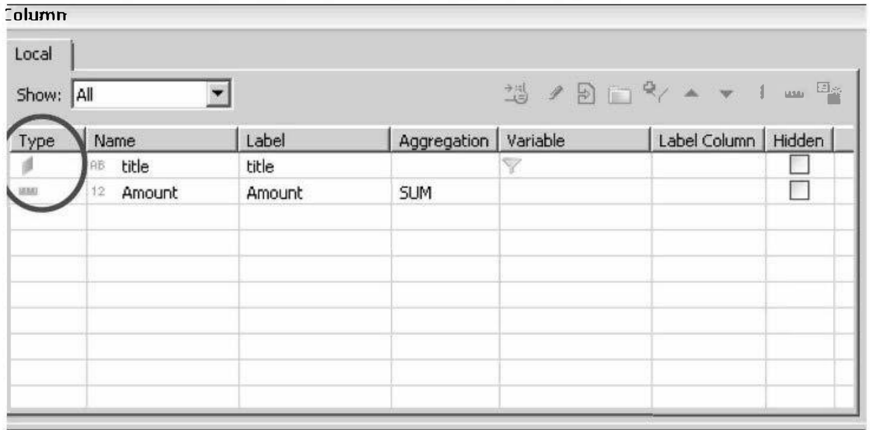


Figure 233: Fields assigned as type Attribute or Measure

r) You now have to save activate your calculation view. You can press the save button, or press Control-S. Then right-click on your calculation view in the Systems tab and select Activate.

Or you can simply press the **Save and Activate** button on the toolbar above the calculation view.

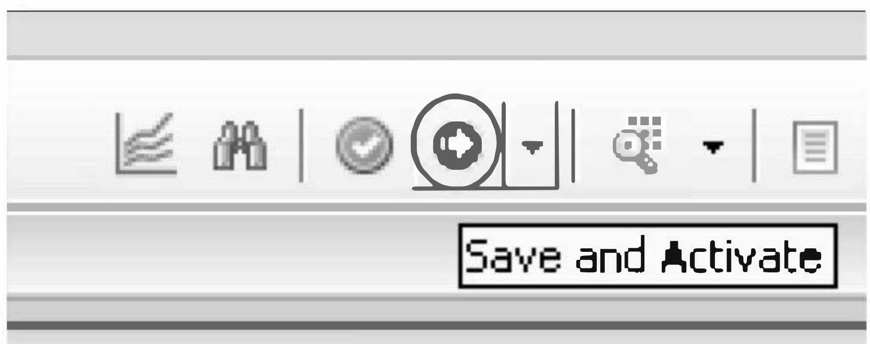


Figure 234: Save and activate

s) Your calculation view should activate successfully.

Continued on next page

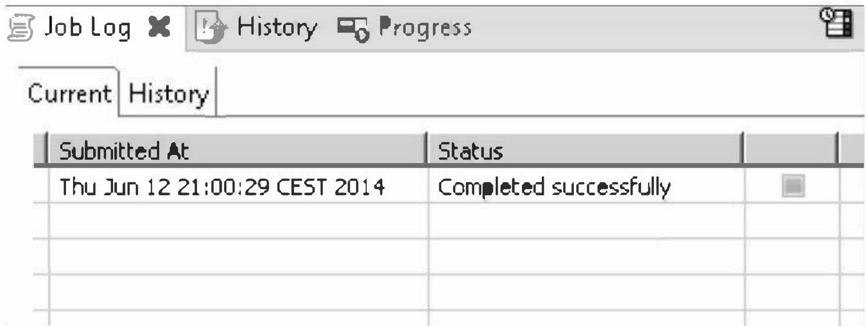


Figure 235: Activated successfully

t) You can preview the data by simply click on the **Data Preview** button on the toolbar above the calculation view,

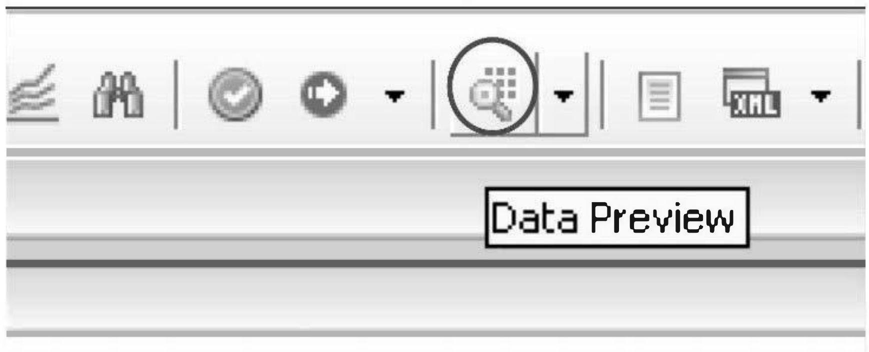
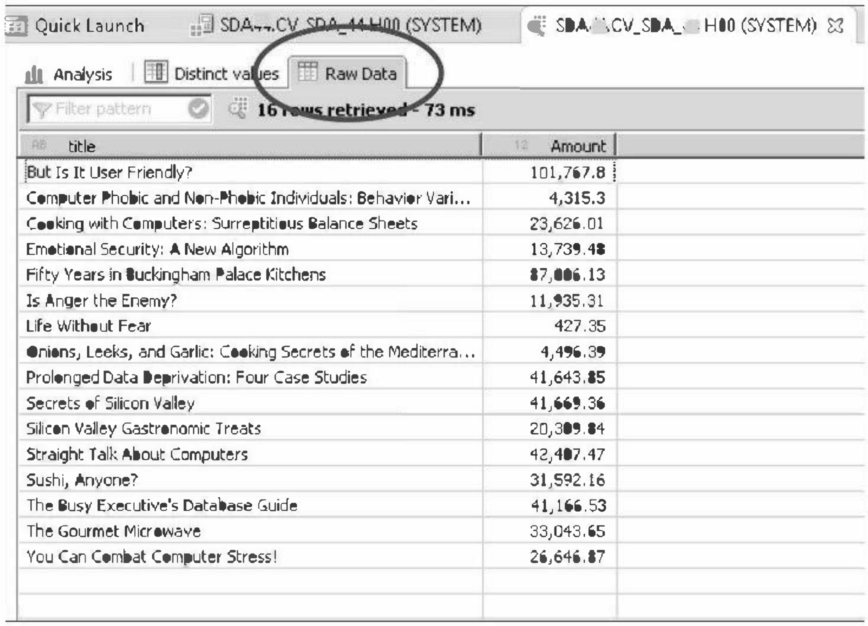


Figure 236: Data Preview of the calculation view

u) Click on the **Raw data** tab to see the actual data.

Continued on next page



title	Amount
But Is It User Friendly?	101,767.8
Computer Phobic and Non-Phobic Individuals: Behavior Vari...	4,315.3
Cooking with Computers: Surreptitious Balance Sheets	23,626.01
Emotional Security: A New Algorithm	13,739.48
Fifty Years in Buckingham Palace Kitchens	87,006.13
Is Anger the Enemy?	11,935.31
Life Without Fear	427.35
Onions, Leeks, and Garlic: Cooking Secrets of the Mediterra...	4,496.39
Prolonged Data Deprivation: Four Case Studies	41,643.85
Secrets of Silicon Valley	41,669.36
Silicon Valley Gastronomic Treats	20,309.84
Straight Talk About Computers	42,407.47
Sushi, Anyone?	31,592.16
The Busy Executive's Database Guide	41,166.53
The Gourmet Microwave	33,043.65
You Can Combat Computer Stress!	26,646.87

Figure 237: Preview the raw data

v) Now click the Analysis tab. Drag the title field to the Labels axis area, and the Amount field to the Values axis area. Now select the treemap graphic view on the toolbar as indicated. Enjoy your report!

Figure 238: Create a treemap graphic in the SAP HANA studio.

Continued on next page

Result

Well done. You have successfully created a virtual table, used it in a calculation view, and generated a useful sales report with data from an external SAP ASE database combined with data in SAP HANA.

Exercise 4: Setting up and using Smart Data Access

Exercise Objectives

After completing this exercise, you will be able to:

- Create a new remote data source
- Create a new virtual table
- Use a virtual table in a calculation view

Business Example

You wish to consume data from **a remote data source** in a calculation view.

Task 1:

You will connect to a remote SAP HANA data source. (As we do not have another SAP HANA data source on this training system, we will just connect back to our own SAP HANA system for the purposes of this training exercise.) To accomplish this, you first create a Smart Data Access connection to our SAP HANA system.

1. Create a Smart Data Access connection to our SAP HANA system using the following connection information:

Parameter	**Value**
Source Name Mode	LOCAL-##
Adapter Name	HANA (ODBC)
Server	wdflbmt7215.wdf.sap.corp
Port	30015
Credentials Mode	Technical User
User Name	SYSTEM
Password	W3lcome123

Continued on next page

Task 2:

Check the relevant configurations and monitoring screens by opening the **Smart Data Access** area in the **Data Provisioning** folder.

1. Open the Smart Data Access monitoring tab, and check the relevant INI file settings in the Administration Console

Task 3:

You will create a virtual table in SAP HANA which refers to a table or view in the remote data source. When you open this virtual table, SAP HANA will fetch the contents from the remote data source.

Create a new schema called SDA##, and add the SFLIGHT table from the remote source as a virtual table.

1. Create a new schema called SDA## (where ## is your student number). Assign the necessary SQL read privileges to the _SYS_REPO user.
2. Now add the "SFLIGHT" table in the "FLIGHT" schema of the remote system as a virtual table. Press F8 or click the Execute button on the toolbar to execute this statement.

Task 4:

You now wish to view the data in your virtual table. Find your virtual table and preview the data

1. Open your SDA## schema, find the LOCAL-##_SFLIGHT table and do a data preview

Task 5:

Create a calculation view where you will use your virtual table together with an SAP HANA table.

1. Create a new calculation view called CV_SDA##
2. Add a Union node to your calculation view. Add your virtual table and the SFLIGHT table to the union node. Map the two tables, set the default client to 'Cross Client', and activate the view. Then preview your data.

Result

The system reads data from a remote source and combines this with data from SAP HANA. You can now use the calculation view for your reporting or analysis.

Solution 4: Setting up and using Smart Data Access

Task 1:

You will connect to a remote SAP HANA data source. (As we do not have another SAP HANA data source on this training system, we will just connect back to our own SAP HANA system for the purposes of this training exercise.) To accomplish this, you first create a Smart Data Access connection to our SAP HANA system.

1. Create a Smart Data Access connection to our SAP HANA system using the following connection information:

Parameter	Value
Source Name Mode	LOCAL-##
Adapter Name	HANA (ODBC)
Server	wdflbmt7215.wdf.sap.corp
Port	30015
Credentials Mode	Technical User
User Name	SYSTEM
Password	W3lcome123

a) Open the SAP HANA studio, select the H00 system in the Systems tab, and open the **Provisioning** folder

b) Right-click on **Remote Sources**, and select **New Remote Source**.

Continued on next page

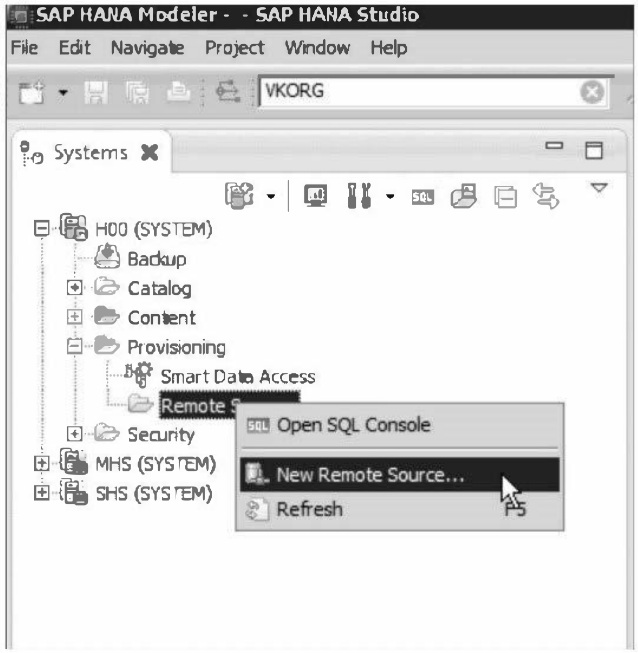


Figure 239: Create a new remote source

c) Create the configuration with the data supplied above. The name of the configuration should be **LOCAL-##** (where ## is your student number, a number between 01 and 10). In the "Adapter Name" dropdown on the right side, select "HANA (ODBC)". Fill in the other parameters as per the above table.

Save your configuration (On the top left toolbar, or by pressing Control-S.)

Continued on next page

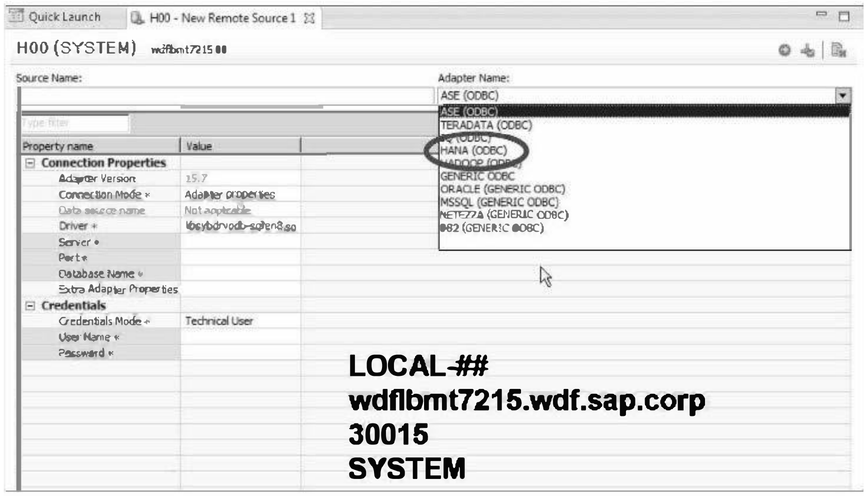


Figure 240: New remote source

d) This is what the configuration will look like once you have saved it.

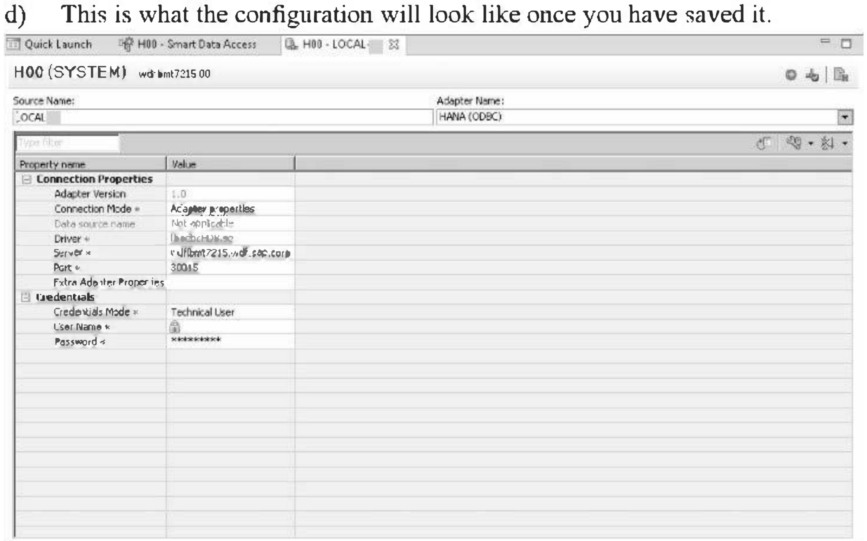


Figure 241: New remote source

e) You have now connected to a Remote Source.

Continued on next page

When you open the Remote Source, you will see the schemas. (In this case we see the same schemas as our HANA system as we are just connected back to our own system for the purposes of this exercise.)

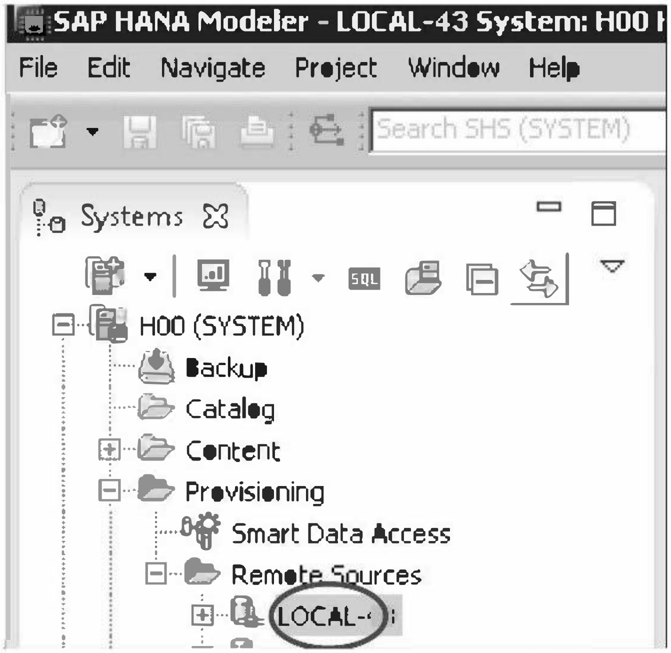


Figure 242: Your new remote data source called LOCAL-##

f)

Task 2:

Check the relevant configurations and monitoring screens by opening the **Smart Data Access** area in the **Data Provisioning** folder.

1. Open the Smart Data Access monitoring tab, and check the relevant INI file settings in the Administration Console

   a) Go to the Provisioning folder of the H00 system. The first option in this folder is the Smart Data Access configuration screen. Double-click on "Smart Data Access" to open this. On the second tab you can monitor your Smart Data Access connections.

Continued on next page

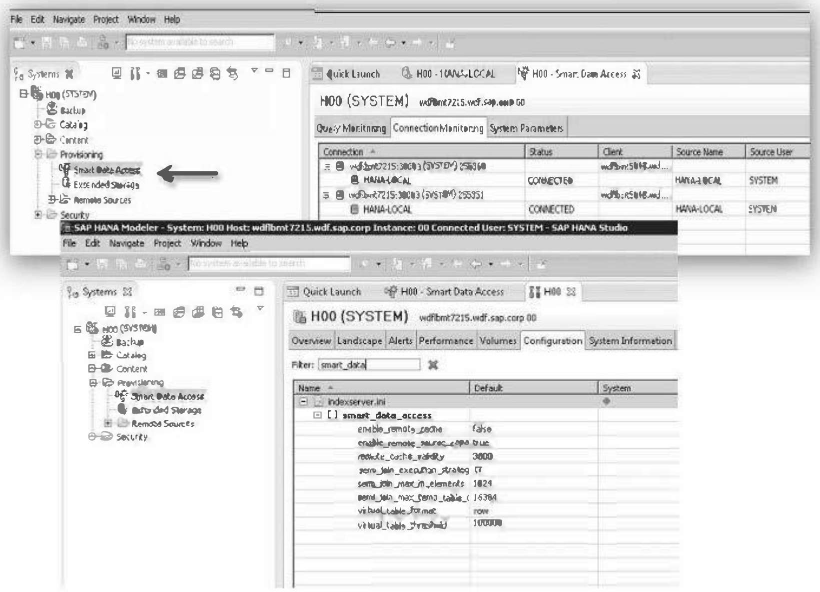


Figure 243: Monitor the connection

b) Go back to the H00 system, and open the Administration Console (by pressing the icon that looks like a screwdriver and spanner)

Continued on next page

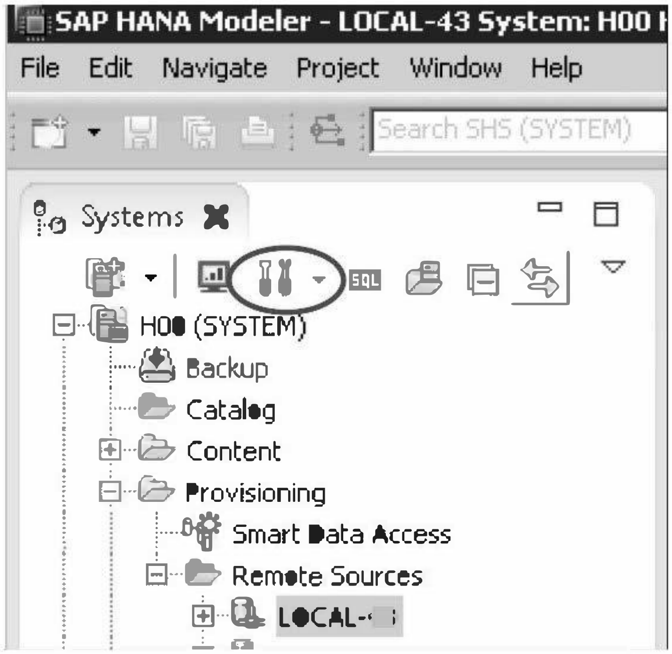


Figure 244: Open the Administration Console

c) In the Configuration tab, type **smart_data_access**. The system will filter the SAP HANA configuration settings and show you only the Smart Data Access settings. You now know where to change it should you need to do this later.

Continued on next page

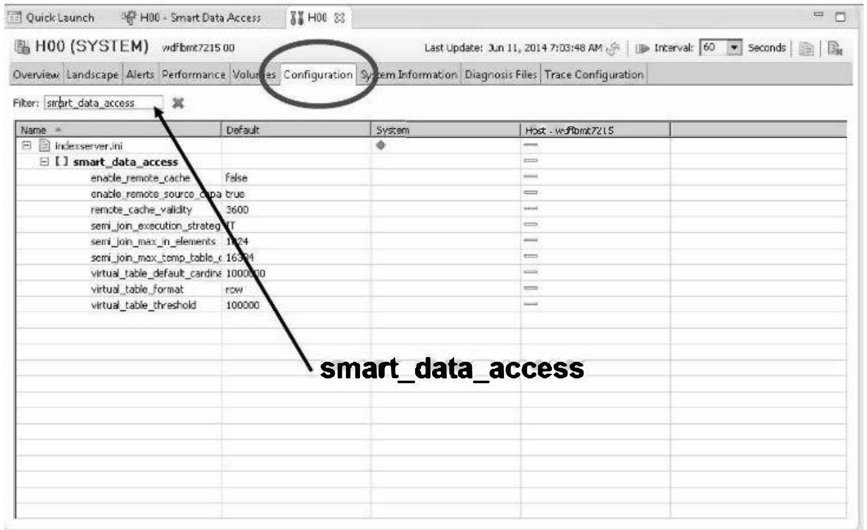


Figure 245: SAP HANA Configuration tab

Task 3:

You will create a virtual table in SAP HANA which refers to a table or view in the remote data source. When you open this virtual table, SAP HANA will fetch the contents from the remote data source.

Create a new schema called SDA##, and add the SFLIGHT table from the remote source as a virtual table.

1. Create a new schema called SDA## (where ## is your student number). Assign the necessary SQL read privileges to the _SYS_REPO user.

   a) Open a SQL Console window. You can do this from the Quick Launch page, or by right-clicking on the H00 system in the "Systems" view".

   b) Type "CREATE SCHEMA SDA##", where ## is your student number.

      You have now created a schema where you can place your virtual table.

Continued on next page

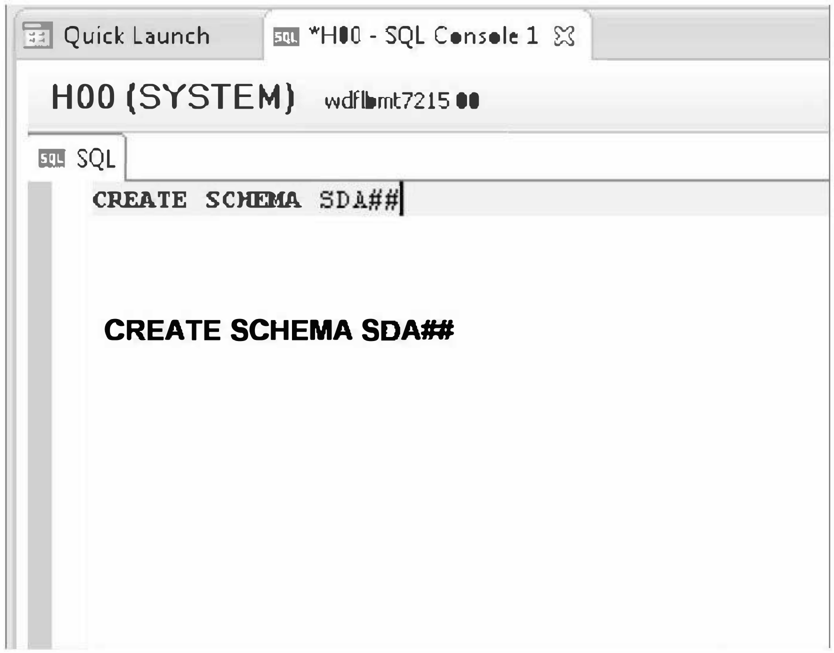


Figure 246: Create your new schema

c) Clear the window, and type "GRANT SELECT on SCHEMA SDA## to "_SYS_REPO" WITH GRANT OPTION;" The ## is your student number. You need to give this privilege, else you cannot use it in your data models. You will get an authorization error during the activation of your data models.

Clear the window again, and type "GRANT SELECT on SCHEMA FLIGHT to "_SYS_REPO" WITH GRANT OPTION". You need to give this privilege, else you cannot use it in your data models. You will get an authorization error during the activation of your data models.

Continued on next page

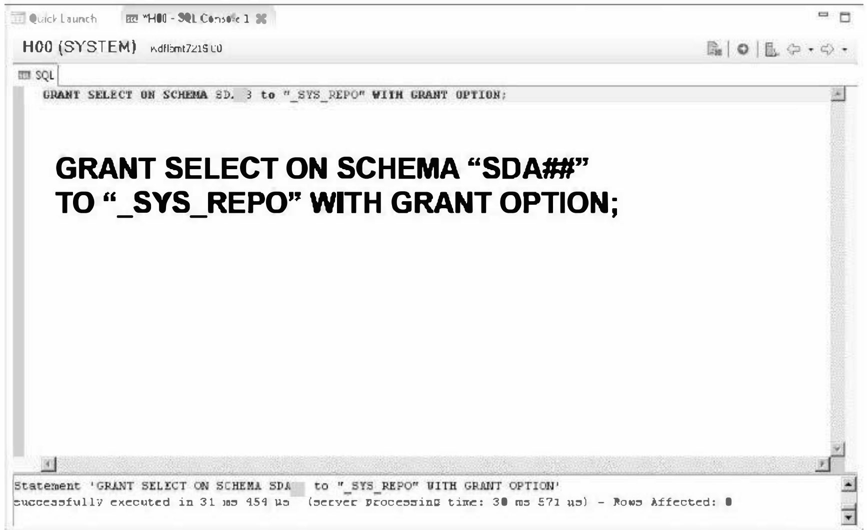


Figure 247: Grant the privileges

2. Now add the "SFLIGHT" table in the "FLIGHT" schema of the remote system as a virtual table. Press F8 or click the Execute button on the toolbar to execute this statement.

   a) Open the "Provisioning" folder in the "Systems" view. Open "Remote Sources". Select your "LOCAL-##" remote source you created.

      Open the "FLIGHTS" schema. Right-click on the **"SFLIGHTS" table**. Select **Add as Virtual Table**

Continued on next page

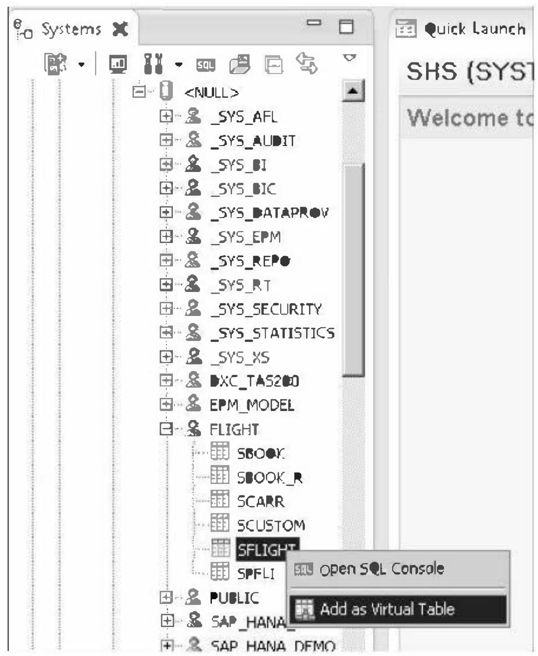
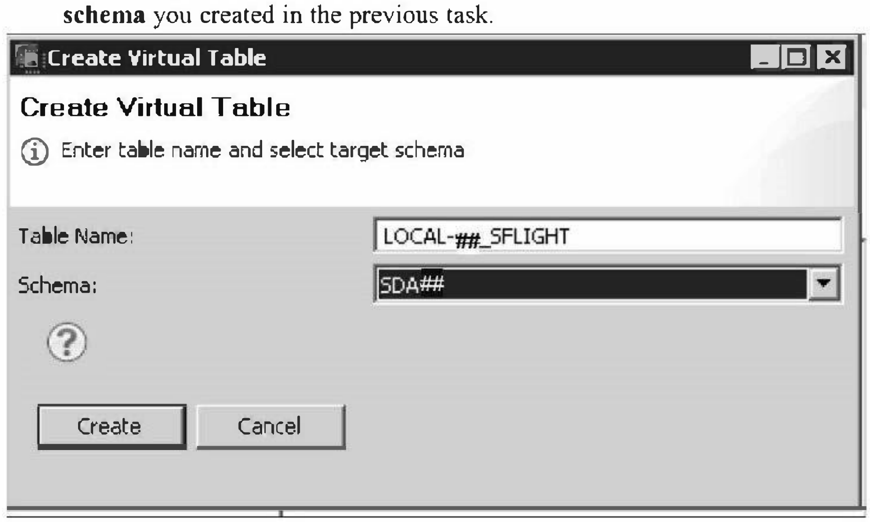


Figure 248: Create a new virtual table

b) Name the virtual table **LOCAL-##_SFLIGHT**. Select your **SDA## schema** you created in the previous task.



Continued on next page

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Figure 249: Name your virtual table LOCAL-##_SFLIGHT

c) You have now created a virtual table!

Figure 250: You created a virtual table!

Task 4:

You now wish to view the data in your virtual table. Find your virtual table and preview the data

1. Open your SDA## schema, find the LOCAL-##_SFLIGHT table and do a data preview

   a) Find your new virtual table by opening the "Catalog" area in the "Systems" view. Select your "SDA##" schema you created. (## is your student number.) Open "Tables"

Continued on next page

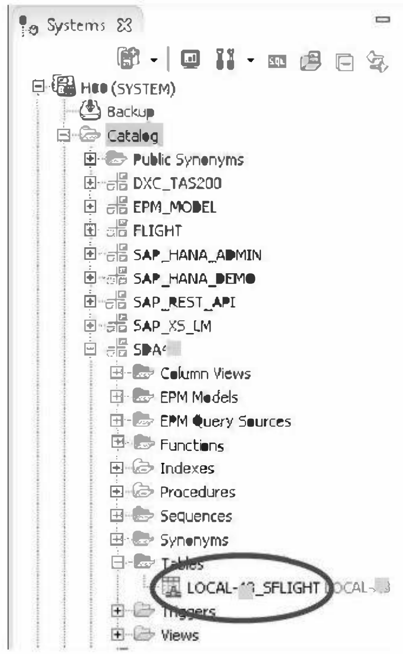
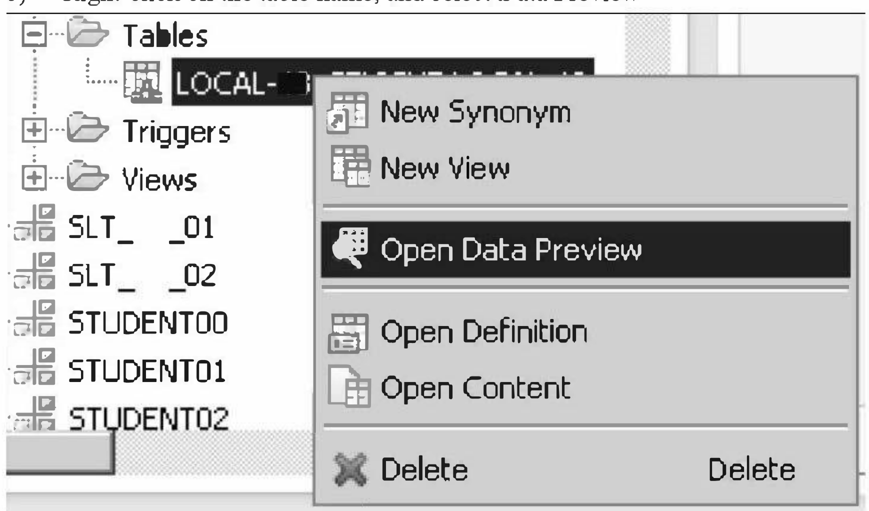


Figure 251: Find your table

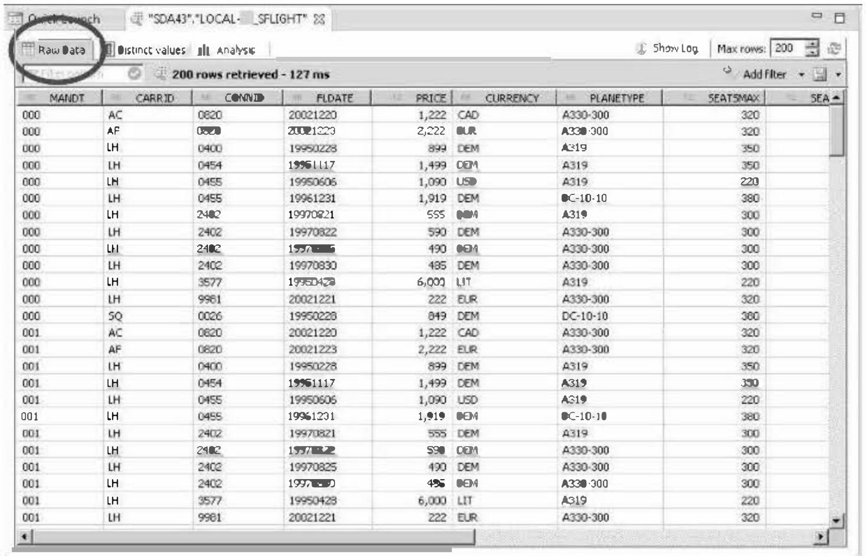
b) Right-click on the table name, and select Data Preview



Continued on next page

Figure 252: Open Data Preview

c) View the data of the virtual table. You are now reading data, stored in a remote source, inside the SAP HANA studio.



MANDT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE	SEATSMAX	SEA
000	AC	0820	20021220	1,222	CAD	A330-300	320	
000	AF	[illegible]	20021223	2,222	EUR	A330-300	320	
000	LH	0400	19950228	899	DEM	A319	350	
000	LH	0454	19951117	1,499	DEM	A319	350	
000	LH	0455	19950606	1,090	USD	A319	220	
000	LH	0455	19961231	1,919	DEM	DC-10-10	380	
000	LH	2402	19970821	555	DEM	A319	300	
000	LH	2402	19970822	590	DEM	A330-300	300	
000	LH	2402	[illegible]	490	DEM	A330-300	300	
000	LH	2402	19970830	485	DEM	A330-300	300	
000	LH	3577	19950428	6,000	LIT	A319	220	
000	LH	9981	20021221	222	EUR	A330-300	320	
000	SQ	0026	19950228	849	DEM	DC-10-10	380	
001	AC	0820	20021220	1,222	CAD	A330-300	320	
001	AF	0820	20021223	2,222	EUR	A330-300	320	
001	LH	0400	19950228	899	DEM	A319	350	
001	LH	0454	19951117	1,499	DEM	A319	350	
001	LH	0455	19950606	1,090	USD	A319	220	
001	LH	0455	19961231	1,919	DEM	DC-10-10	380	
001	LH	2402	19970821	555	DEM	A319	300	
001	LH	2402	[illegible]	590	DEM	A330-300	300	
001	LH	2402	19970825	490	DEM	A330-300	300	
001	LH	2402	[illegible]	495	DEM	A330-300	300	
001	LH	3577	19950428	6,000	LIT	A319	220	
001	LH	9981	20021221	222	EUR	A330-300	320	

Figure 253: View the data

Task 5:

Create a calculation view where you will use your virtual table together with an SAP HANA table.

1. Create a new calculation view called CV_SDA##

   a) Create a new graphical calculation view.

Continued on next page

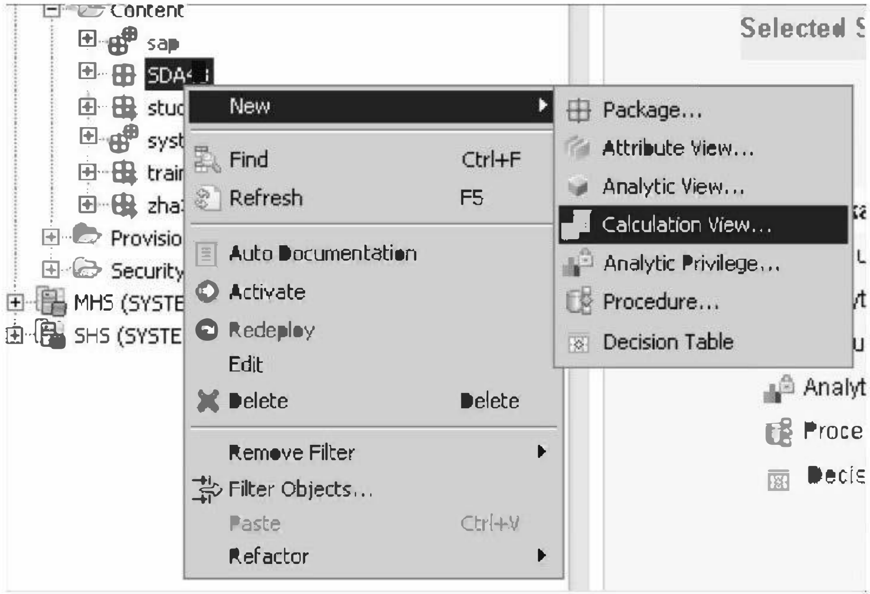


Figure 254: Create a calculation view

b) Name the graphical calculation view ."CV_SDA##", where ## is your student number. Click on "Finish". The "CV_" naming convention is used to indicate that this is a calculation view.

Continued on next page

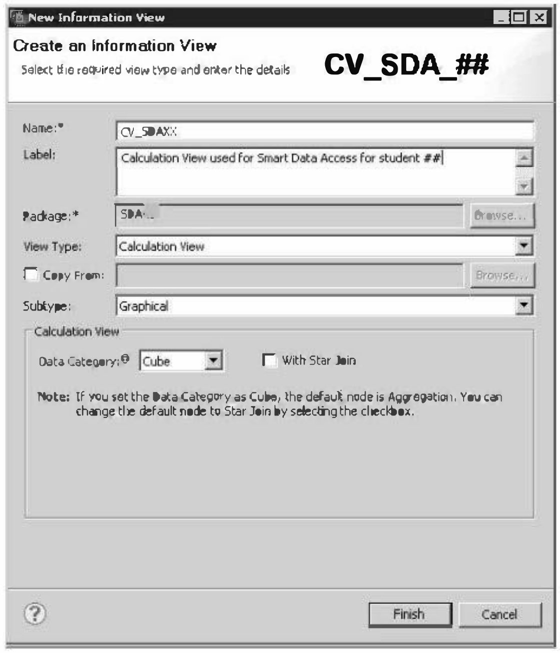


Figure 255: Create the CV_SDA# calculation view

2. Add a Union node to your calculation view. Add your virtual table and the SFLIGHT table to the union node. Map the two tables, set the default client to 'Cross Client', and activate the view. Then preview your data.

   a) Add a Union node below the Aggregation node. Link the Union to the Aggregation node, and select the Union node.

Continued on next page

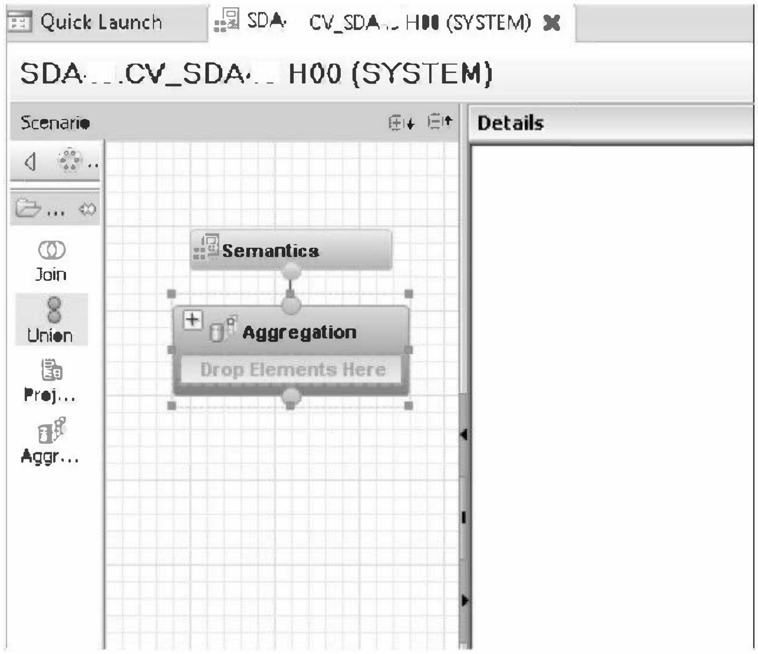


Figure 256: Add a Union node

b) Add the SFLIGHT table from the FLIGHT schema in your SAP HANA system to the Union node.

Continued on next page

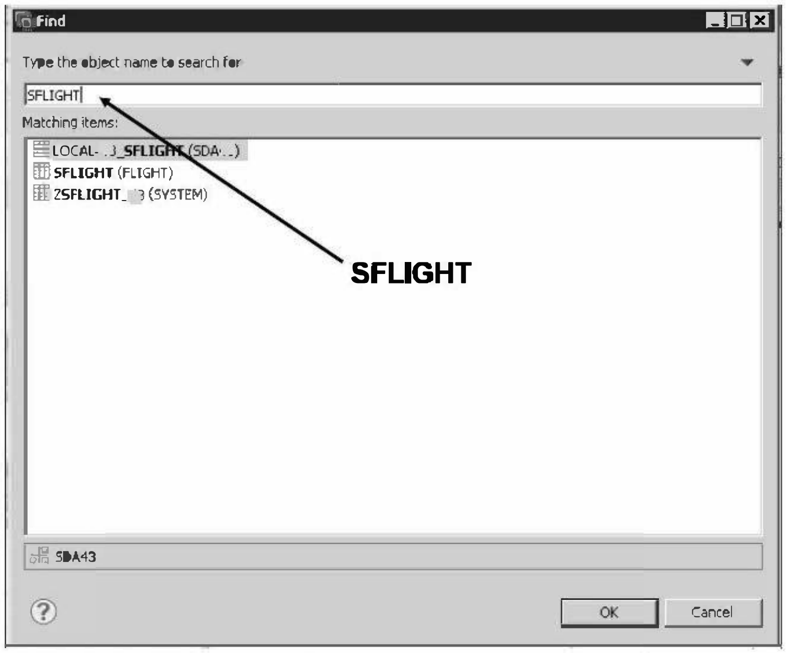


Figure 257: Add the SFLIGHT table

c) From your SDA## schema, add your virtual table to the Union node.

Continued on next page

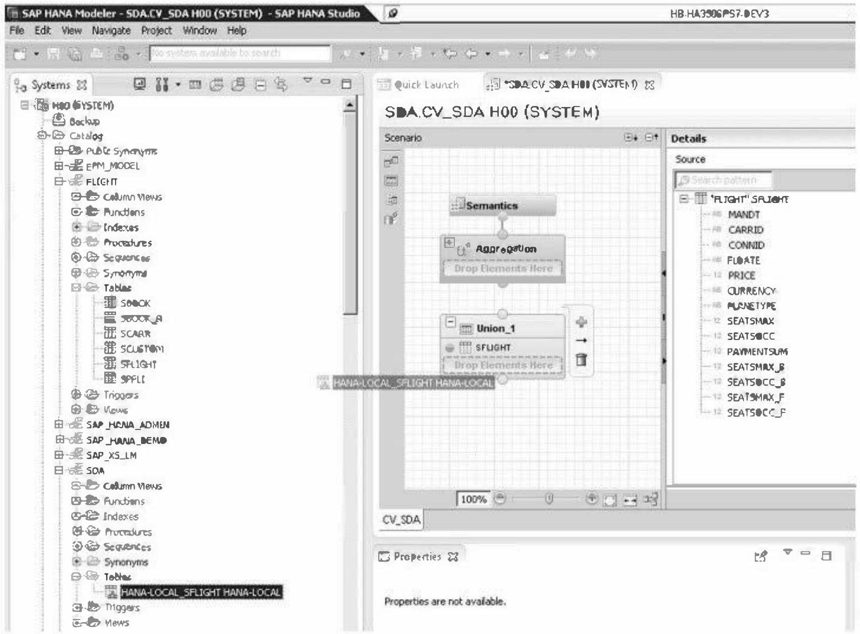


Figure 258: Add virtual table

d) Select the Union node.

Click on the "Auto Map by Name" button on the toolbar. This will automatically map the fields of the local SAP HANA table and of the virtual table.

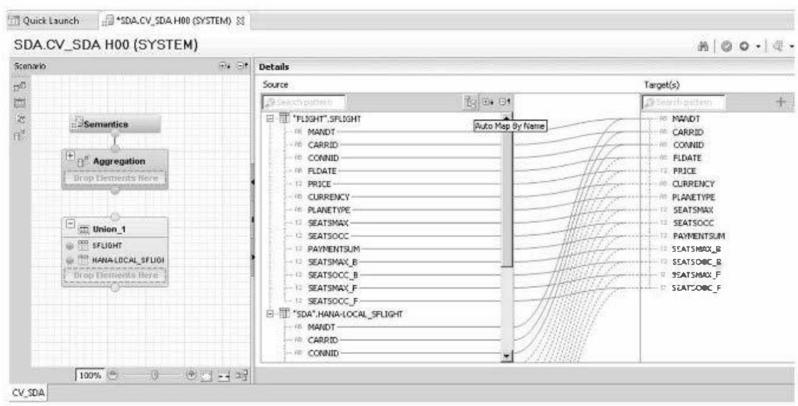


Figure 259: Mapping fields

Continued on next page

e) Select the Aggregation Node

Select all the fields and add them to the output. (Right-click on the top bar, and select **Add All To Output**)

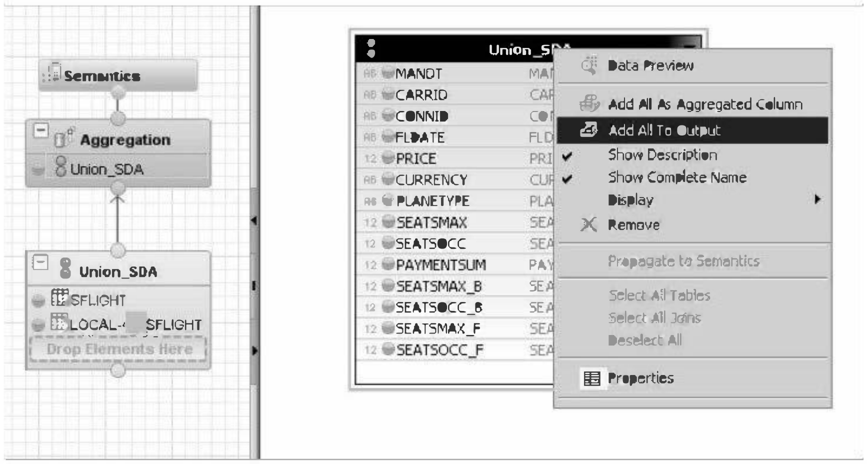
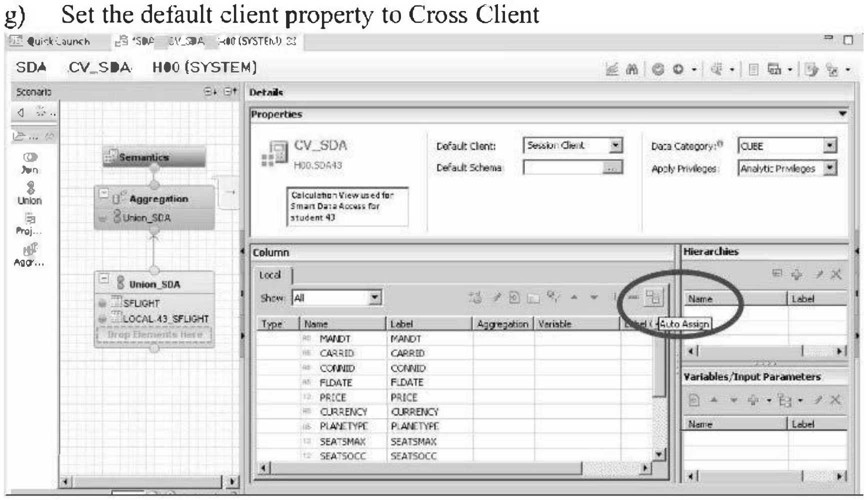


Figure 260: Add all fields to the ouput in the Aggregation node

f) Select the Semantics node.

Click the "Auto Assign" button on the toolbar to let the system assign which fields are measures and which are attributes.

g) Set the default client property to Cross Client



Continued on next page

Figure 261: Auto Assign

h) Save and activate the calculation view.

Hint: If you get an activation error, you probably forgot to assign the privileges earlier. See above where you created the SDA## schema.

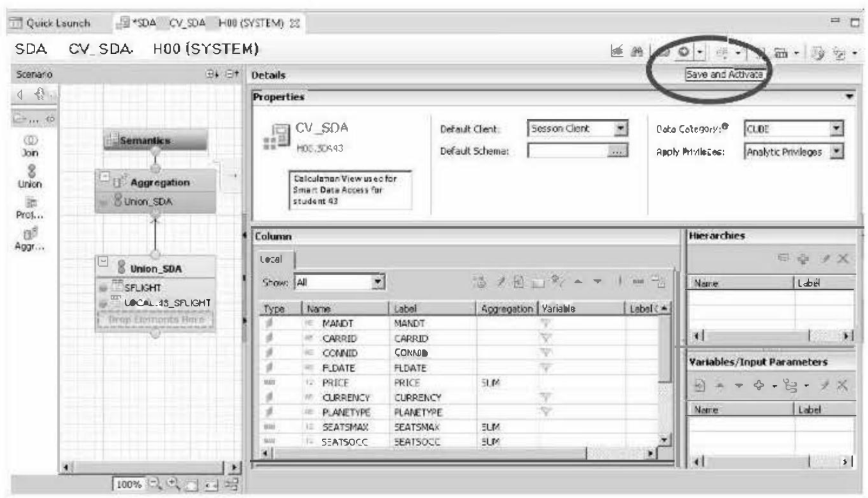


Figure 262: Save and Activate the calculation view

i) Select "Data Preview" of your calculation view.

Continued on next page

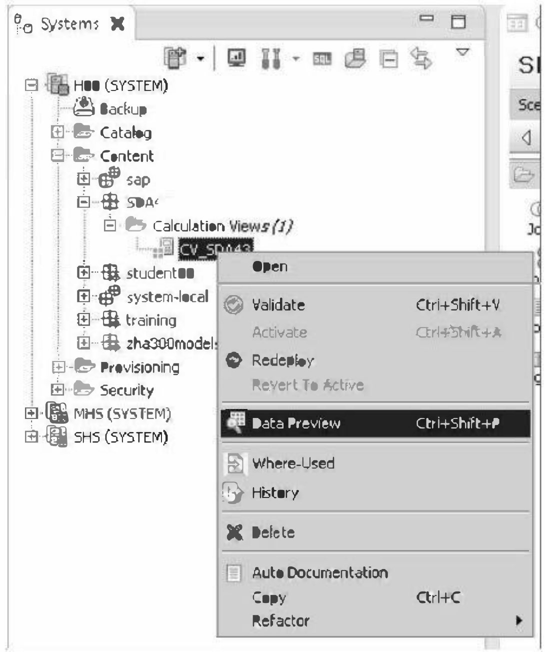


Figure 263: Select data preview

j) You can now view the data from your calculation view.

Continued on next page

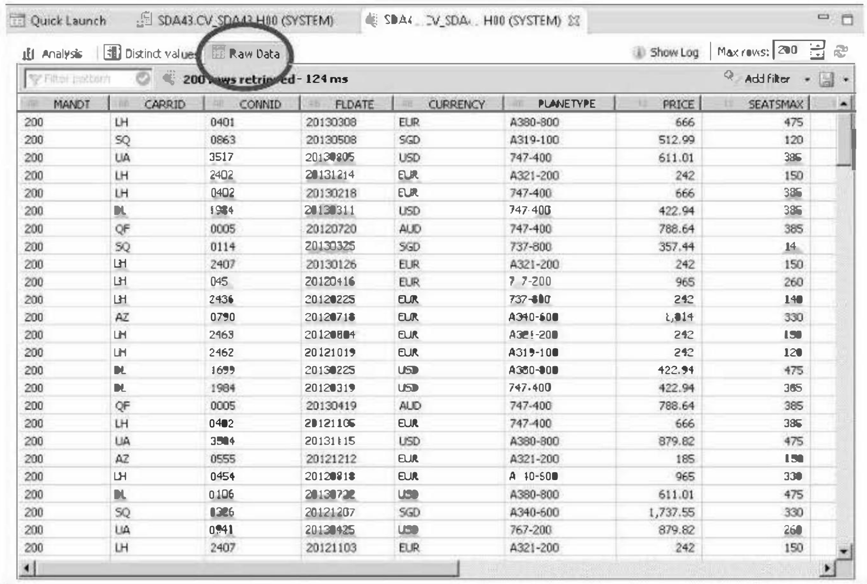


Figure 264: The data

Result

The system reads data from a remote source and combines this with data from SAP HANA. You can now use the calculation view for your reporting or analysis.

Lesson Summary

You should now be able to:

- Understand the Smart Data Access concepts
- Know when to use Smart Data Access
- Know how to implement Smart Data Access
- Build data models using virtual tables.

Unit Summary

You should now be able to:

- Understand the Smart Data Access concepts
- Know when to use Smart Data Access
- Know how to implement Smart Data Access
- Build data models using virtual tables.

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Unit 5

SAP HANA Enterprise Information Management (EIM)

Unit Overview

Unit Objectives

After completing this unit, you will be able to:

- Understand SAP HANA smart data integration (SDI) •
- Understand SAP HANA smart data quality (SDQ)

Unit Contents

Lesson: SAP HANA Enterprise Information Management and its components Smart Data Integration and Smart Data Quality

Lesson Overview

A brief introduction to the new features of Smart Data Integration and Smart Data Quality in SAP HANA.

Lesson Objectives

After completing this lesson, you will be able to:

- Understand SAP HANA smart data integration (SDI) •
- Understand SAP HANA smart data quality (SDQ)

Business Example

Up to now we either imported data into SAP HANA using ETL processes in batch mode (not real-time) if we wanted clean data (data quality), or using SAP LT Replication Server in real-time (with no data cleaning). Using the new Smart Data Integration and Smart Data Quality in SAP HANA, we can leverage the power of SAP HANA to give us both data quality and real-time replication

SAP HANA Enterprise Information Management

What is Enterprise Information Management?

Enterprise Information Management (EIM) enhances, cleanses, and transforms data to make it more accurate and useful.

With the speed advantage of SAP HANA, the new SAP HANA EIM option can connect with any source, provision and cleanse data, and load data into SAP HANA on-premise or in the cloud.

By providing EIM capabilities in HANA, SAP HANA EIM offers:

•A simplified landscape — one environment in which to provision and consume data.

•Access to more data formats, including an open framework for new data sources.

•In-memory performance, which means increased speed and decreased latency.

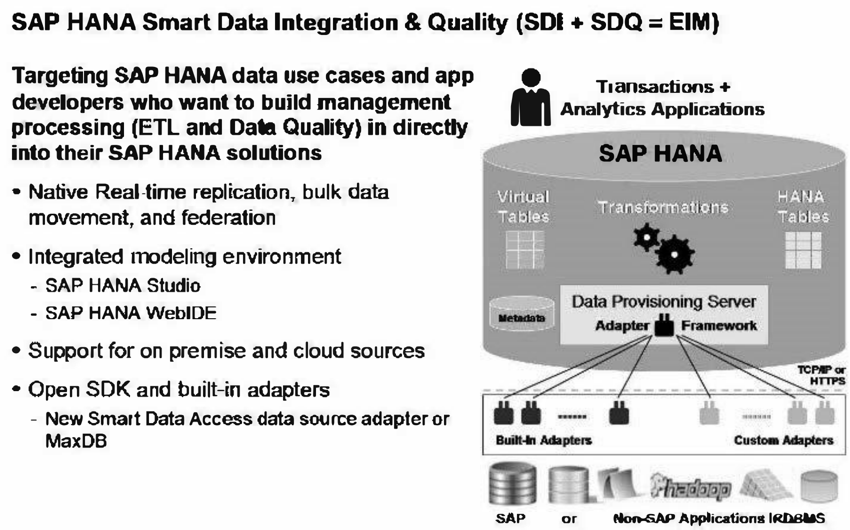


Figure 265: EIM = SDI + SDQ

SAP HANA EIM consists of two main areas, smart data integration and smart data quality, as described below ..

Feature area	Description
Smart data integration	Real-time, high-speed data provisioning, bulk data movement, and federation. SAP HANA EIM provides built-in adapters and an SDK so you can build your own. Smart data integration includes the following features and tools: •Web-based Development Workbench Replication Editor, where you can set up real-time data replication tasks in an easy-to-use web application •Smart data integration adapters, delivered by SAP to connect to sources and enable replication and transformation. •An Adapter SDK, to create custom adapters •Enhanced smart data access functionality for connecting to remote sources in real time •Additional monitoring interface in the SAP HANA Cockpit for monitoring Data Provisioning Agents, remote subscriptions, and tasks •Data Provisioning Agent, which enables adapters and connecting remote sources to SAP HANA

	•Application Function Modeler (AFM) nodes to perform common data provisioning tasks, such as pivot, history preserving, lookup, and table comparison. (AFM is a modeling tool available within SAP HANA Studio that performs predictive analysis in a user-friendly interface.)
Smart data quality	Real-time, high-speed data cleansing, address cleansing, and geospatial data enrichment. SAP HANA EIM provides an intuitive interface to define data transformation flowgraphs in SAP HANA Studio. Smart data quality includes AFM nodes to perform data quality tasks, such as address cleansing, data cleansing, and geocoding.

You can replicate or/and transform data (batch and real time) on the way to the SAP HANA system (on premise or in the cloud). Transformation of the data could include cleansing data, through smart data quality

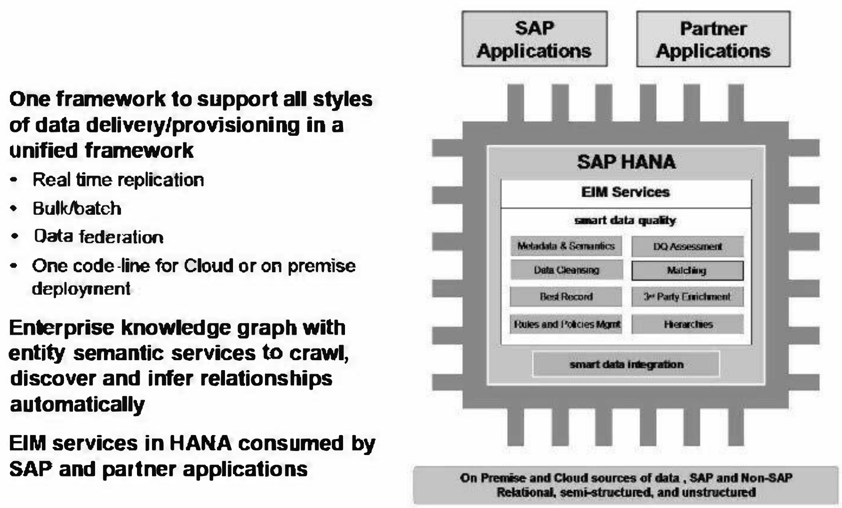


Figure 266: EIM Services

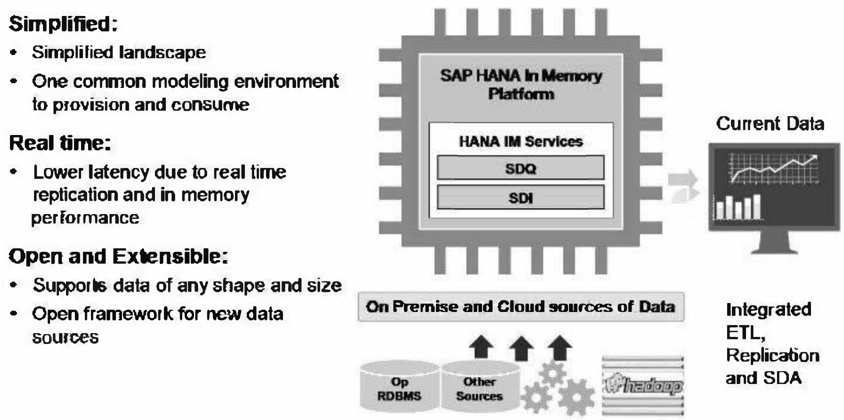


Figure 267: SAP HANA EIM Services - Simplifying the Landscape and Lowering Data Latency

Smart Data Integration

To accelerate the reporting performance with disk-centric computing, pre-aggregates were required, which led to the need for separate OLTP system and OLAP systems. In addition, an ETL system is then required to transform data from the OLTP system to the OLAP system overnight, making results available for analysis and reporting after around 24 hours from the time data has been collected.

The SAP HANA SDI has the following properties:

- It provides both data replication and data transformation services,
- It is open and extensible,
- It works on SAP and non-SAP data of any style, shape and size,
- Its modeling environment is part of HANA Studio and HANA Web-based Development Workbench,
- It extends HANA's transformation capability by integrating ETL-type transformations natively,
- It accelerates performance through a native HANA implementation.

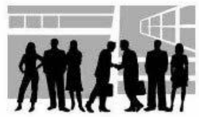
Lower TCO	Open & Extensible	Real-time
		
• Simplified Landscape, Integrated modeling environment • Single product covering multiple use cases • Build once; deploy on premise or on cloud	• Open framework • Data – any style, shape and size- • SAP and non-SAP	• Ability to replicate and transform data in real time • Transactional consistency and guaranteed delivery • Breakthrough Performance (natively built in HANA)

Figure 268: SAP HANA EIM Services – Value Proposition

- Cloud integration made simple and reliable – process and data integration, real-time, HANA optimized
- Faster time-to-value with prebuilt integration solutions for on-premise and cloud applications from SAP and others
- Expertise you can trust – established mass of SAP integration customers, vibrant SAP Community Network, partner network with product knowledge
- Real-time innovation – Extract and load data natively into SAP HANA
- Scale your integrations to match the most demanding enterprise needs – proven power of HANA ASE reliant messaging, Industry's best maintenance and support offerings

SAP HANA Smart Data Integration (SDI)

- Provides real time push mode to replicate sources
  - On selected sources with change data capture (CDC) capability
  - Leverages proven Sybase Replication Server technology
- Provides *batch pull mode* for all types of sources
- Interacts well with HANA's federation technology (SDA)
  - Leverages SDA anytime remote data is requested (initial data load, queries, etc).
  - Extends SDA connectivity by providing an Adapter SDK
- Provides Transactional Integrity for real time push
  - By listening to changes in the DBMS transaction logs and only replicating committed changes
- Provides Guaranteed Delivery for real time push
  - SDI can resume processing if replication stream is halted or disrupted
  - SDI can continue to operate during a temporary absence of the HANA target

Figure 269: Smart Data Integration - Extending HANA by Integrating Real Time Delivery Mode

Extends HANA's capability by integrating ETL-type transformations natively in HANA

Transformation inputs include:

- HANA tables
- Virtual HANA tables and
- Views

For definition of transformation flows (in Batch and in real time) you can use the Application Function Modeler (AFM) which is a part of SAP HANA Studio .

Available transformations

- **Basic SQL – oriented**
  Filter, Join, Union, Sort, Map
- **Advanced SQL – oriented**
  Aggregation, lookup, sort, case, and pivot/unpivot
- **Addressing the data movement lifecycle**
  Row generation, date generation, table comparison and history preserving
- **Executing code**
  Script, function, procedure

Figure 270: Available Transformations

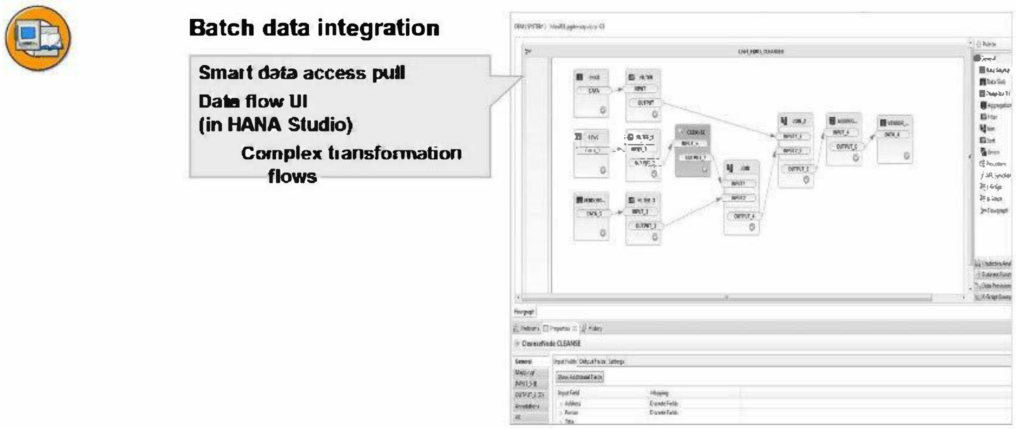


Figure 271: Batch data integration

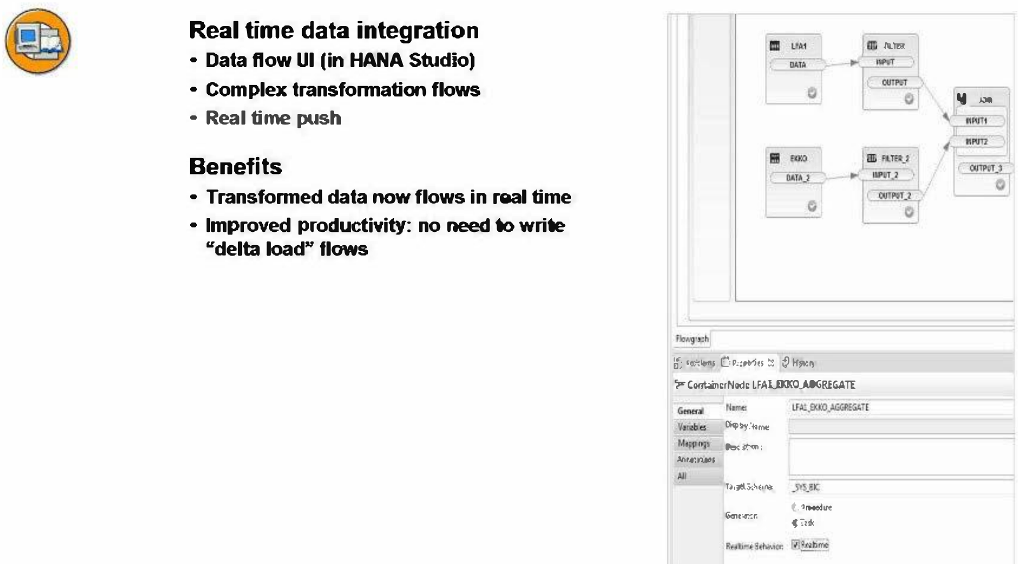


Figure 272: Real time integration

A task is a long running program executing transforms

Task has a plan that compiles to a HANA calculation scenario

- **Operations in plan may be implemented by existing calculation nodes or by IM-specific native nodes**

Runs in the background

Needs monitoring

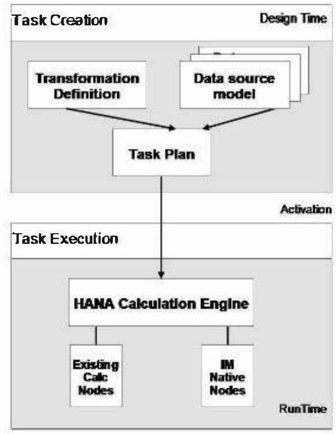


Figure 273: Native HANA Implementation of Transformation as Tasks

Real Time Adapters

- Log Reader adapters
  - **For Oracle, MSSQL and DB2**
- SAP ECC adapters
  - Implements log reader adapter functionality on top the same DBMS sources as above.
  - In addition, it takes into account:
    - SAP ABAP Transparent and Clustered tables
    - SAP catalog hierarchy metadata
- Twitter adapter
  - **Real time access to Twitter content on hashtags, users, etc.**

Batch Adapters

- OData adapter
  - Adapter for oData services
  - Generally accessible from the cloud (e.g., Success Factors)
- Hive adapter
  - Adapter for Hadoop

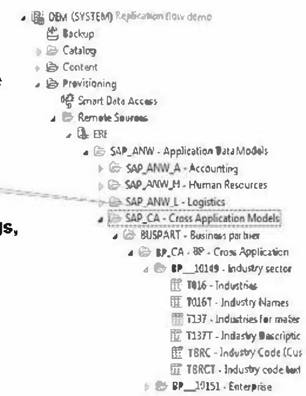


Figure 274: Smart Data Integration – Built-in Adapters for Common Sources

You can see the extensive list of real-time and batch adapters that are available.

Leveraging the power of SAP HANA, we can now deliver quality data faster than before.

Remote subscriptions

- Source and target, status, last processed transaction time…
- General statistics

DP Agent monitor

- Status/statistics on agents and on adapters within Agent

Task monitor

- Task complete (yes/no), time it took
- Statistics within a task
  - Percentage completed
  - number of records transformed,
  - Transforms executed

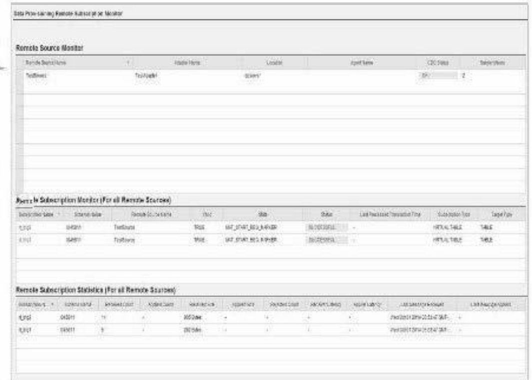


Figure 275: Monitoring

Smart Data Quality (SDQ)

- **Transformations enriching data**
- **Cleanse**

  Parse, standardize and enrich person, title, phone, firm, email and address information within a specified input source.
- **Geocode**

  Enrich address data with associated latitude and longitude information

---

- **Simplify cleanse transforms:**
  Single transform deals with
  - Person names and titles, phone, email,
  - Firm,
  - Address information

  In Data Services, it is in 2 transforms
- **Consolidate available configuration options:**
  Improved productivity at functional parity

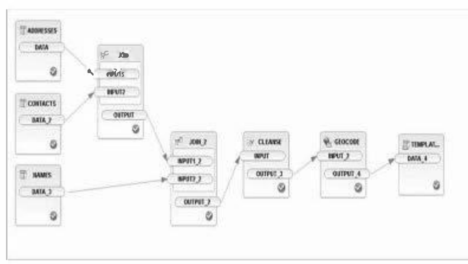


Figure 276: Smart Data Quality - Transformations

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Lesson Summary

You should now be able to:

- Understand SAP HANA smart data integration (SDI) •
- Understand SAP HANA smart data quality (SDQ)

Unit Summary

You should now be able to:

- Understand SAP HANA smart data integration (SDI)
- Understand SAP HANA smart data quality (SDQ)

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Unit 6

Smart Data Streaming

Unit Overview

Unit Objectives

After completing this unit, you will be able to:

- Understand SAP HANA smart data streaming
- Know where to use Smart Data Streaming
- Understand the value of Smart Data Streaming in certain use cases

Unit Contents

Lesson: Smart Data Streaming

Lesson Overview

A quick introduction to new functionality in SAP HANA to capture, process, and react to real-time event (data) streams.

Lesson Objectives

After completing this lesson, you will be able to:

- Understand SAP HANA smart data streaming
- Know where to use Smart Data Streaming
- Understand the value of Smart Data Streaming in certain use cases

Business Example

We discuss a few business or use cases in this short chapter.

Introduction to Event Stream Processing

Event Stream Processing

- Extends the capabilities of the SAP HANA Platform with the addition of real-time event stream processing
- Capture data arriving continuously from devices and applications
- Act on new information as soon as it arrives: alerts, notifications and immediate response to changing conditions
- Stream live information to operational dashboards
- Highly scalable - Process hundreds of thousands or even millions of events per second

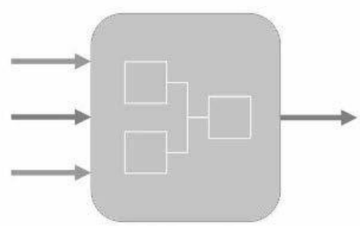


Figure 277: SAP HANA Smart Data Streaming - A New Component with SPS09

Welcome to Event Stream Processing (ESP). This product has been a stand-alone product from SAP, and before that from Sybase, for a while. It is now part of SAP HANA, and extends the capabilities of the SAP HANA platform with the addition of real-time event stream processing. We can capture data (millions of events per second

) arriving continuously from devices and application, act on this new information as soon as it arrives, and react using alerts, notifications and immediate response to changing conditions. With SAP HANA, it is called Smart Data Streaming.

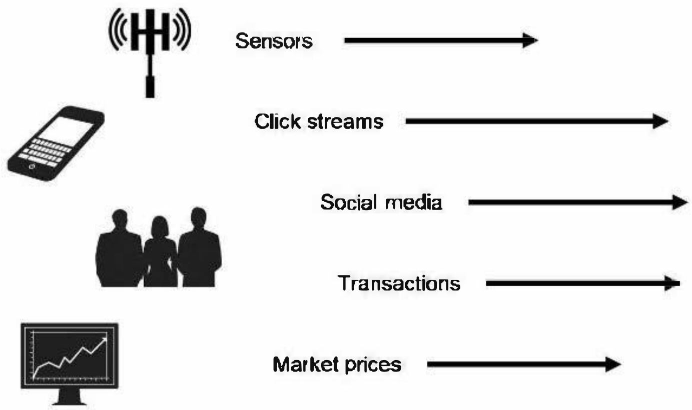


Figure 278: SAP HANA Smart Data Streaming - Streaming Data Sources Are Becoming Ubiquitous

Enterprises today are flooded with streams of messages as things happen. Individual events may not be significant by themselves, but how do you extract insight from the noise? How do you know when something of significance has occurred? You might have thousands of sensors reporting status every few seconds – and most of that information is uninteresting. But when something is starting to go wrong, you want to know as soon as possible, so that you can take action before a small trend becomes a big problem.

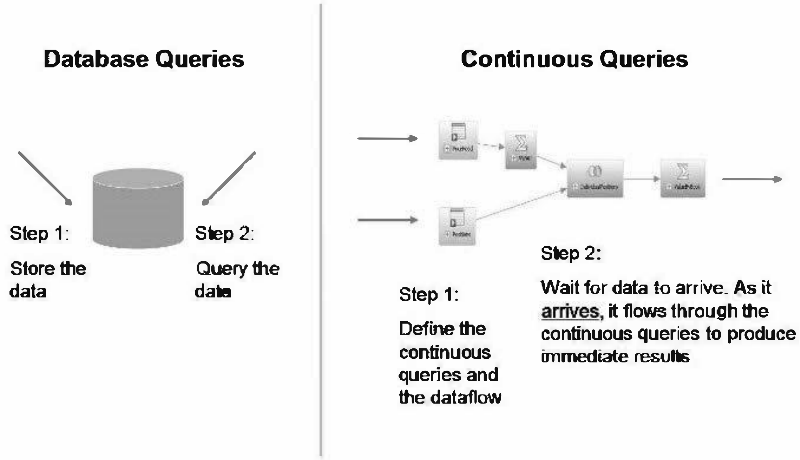


Figure 279: SAP HANA Smart Data Streaming - Database Queries vs. Continuous Queries

SAP HANA smart data streaming extends the capabilities of the SAP HANA Platform

As we talk about how ESP applies continuous queries to streams of events, and how you can collect events in windows so that you can examine new events in the context of past events, and this can be confusing because it starts to sound like a database. But it's important to understand that ESP is *not* a database.

With ESP you define CONTINUOUS queries. These are not queries in the traditional sense that are run against a database. These queries are defined in advance and they are event-driven, updating continuously as new information arrives. And while ESP can hold data temporarily in windows, it does not provide a way to store data permanently. that is obviously where SAP HANA comes in.

And while there is some overlap in the types of analytics you can apply to the data, probably the most basic distinction is this: ESP is event-driven – it publishes output in response to the arrival of new event information. Whereas a database operates on-demand: it delivers results in response to queries that are run against it by a user, by an application, or on a scheduled basis.

Now it's true, that most databases have triggers, that let you define an action to take in response to new information being added to the database. But triggers are much more constrained in what they can do, and they act in response to a single event – whereas ESP can act based on combinations of events, such as trends or patterns. There's also the streaming architecture of the ESP engine, which is designed to stream new events through the continuous queries at very high rates – hundreds of thousands of incoming events per second, streaming through the continuous queries to produce new output – something database triggers just weren't designed for.

And finally, there's the aspect of streaming output. When you want to see what data has changed in a database, you need to query it. To monitor a database for changes, you need to regularly poll it. ESP on the other hand, produces streaming output: it will tell you when something changes.

So bottom line, is that ESP is distinctly different from – and complementary to – a database – event a high-speed real-time database like HANA. It can't do what a database does, and where streaming data is concerned, it provides capabilities that a database can't provide.

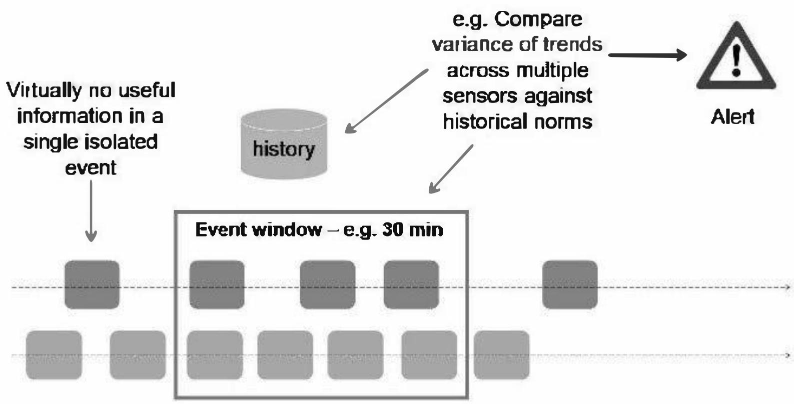


Figure 280: SAP HANA smart data streaming - Extracts insight from events

So how does ESP work? ESP lets you define continuous queries that are applied to one or more streams of incoming messages to produce streams of output events. The output events are the "complex events" we're talking about when we talk about complex event processing (CEP). These are higher level events that are produced by ESP on the basis of the analysis of the incoming events.

This is what sets ESP (or CEP) apart from simple event processing. It's the ability to examine incoming events in the context of other events or other information, to understand what's happening. In many cases, a single event may not contain much information or by itself be interesting, but when combined with other events, you may be able to observe a trend or pattern that is very meaningful.

Let's take a simple example: let's say you want to watch for some overheating equipment. You've got sensor data coming in that tells you the equipment temperature. But that data alone might not be very interesting. If the equipment temperature is 90 degrees, but it's outside and the outside air temperature is 85 degrees, that might be normal. But if the equipment is 90 degrees when the air temp is 30 degrees, that could indicate a problem.

Or maybe it's a trend you care about: so an individual data point doesn't tell you much, but you want to look at how it has changed in the last hour – or the last 5 seconds. Or maybe you want to compute a moving average and compare it to historical norms for similar equipment, or users, or context.

These are the types of things you can do with ESP.

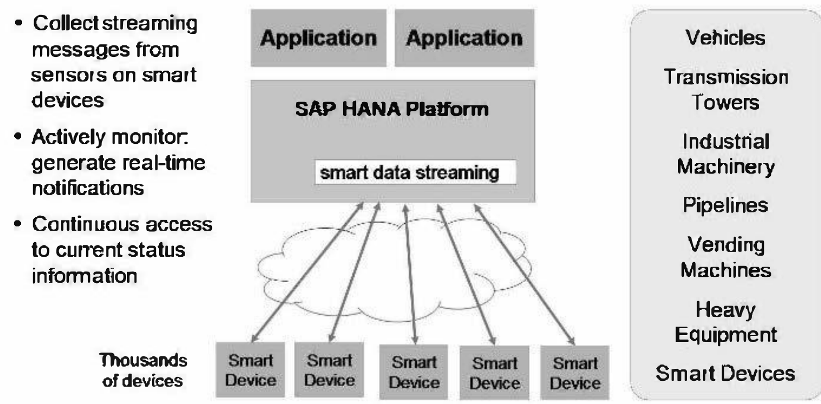


Figure 281: SAP HANA Smart Data Streaming -Use case: Active Equipment Monitoring

The Internet of Things is not the future. It's already here. Millions of devices are now connected and delivering information. We even have lightbulbs with built-in WiFi.

But how do you take advantage of all that data that's available? How do you collect it an analyze it? And what if you want to be able to respond immediately to what's going on?

That's where ESP comes in: it's ideally suited to collecting this data from thousands of devices – not after the fact but as fast as they send the data. But it doesn't stop at data collection. The real value comes from the ability to actively monitor and understand this data. Take advantage of short-lived opportunities. Or act on emerging problems before they become big problems.

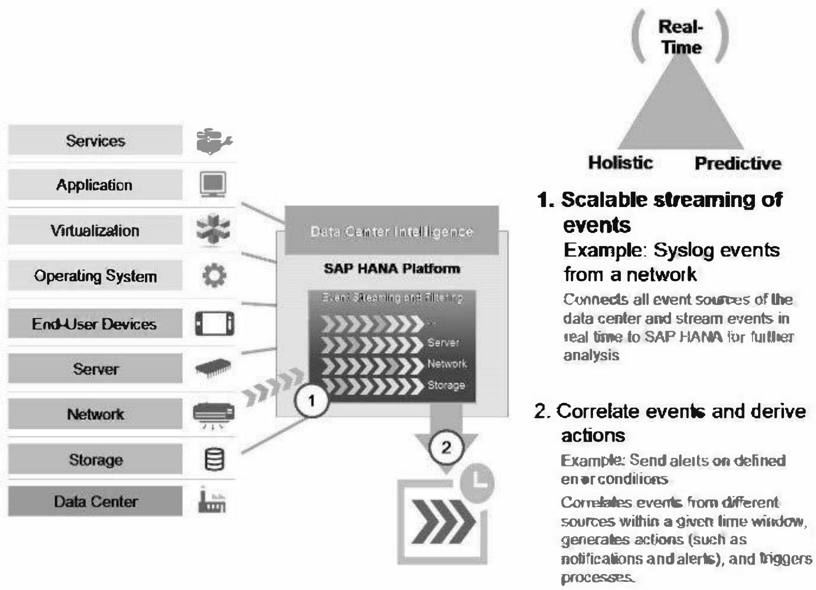


Figure 282: SAP HANA Data Center Intelligence - Use case: Real-time

Regarding the real-time use case, we talk about

- The scalable streaming of event. As an example let's use the syslog events from the network and stream these events into the event store for further analysis.
- Correlation of events and derive actions. An example here is the notification on alerts on defined error conditions by correlation of event from different sources within a given time window.

Business Challenges

- Maintain a fair and honest trading environment
- Provide a highly competitive trading platform
- Integrate with international markets

Technical Challenges

- Detect abusive trading patterns in real-time – at high speed market rates
- Requires a solution that can be rapidly deployed
- Ability to integrate multiple data sources
- Easy to deploy new/updated monitoring algorithms

Benefits

- Allowed for rapid development and deployment
- Enables a fair trading environment
- Reduced operational costs of surveillance and trade monitoring function

Figure 283: SAP HANA Smart Data Streaming - Use Case: Capital Markets

Trading surveillance applications take many forms and go beyond market surveillance undertaken by exchanges. Many brokers rely on surveillance to ensure that orders going through their systems comply with all exchange and market regulations and dark pool operators use surveillance to prevent information leakage.

- Streaming "projects" define the input streams, continuous queries, and outputs
- Projects are defined in CCL (Computation Continuous Language), a SQL-based stream processing language
- Streaming plug-in for SAP HANA studio provides both a CCL editor and visual editor, along with testing tools

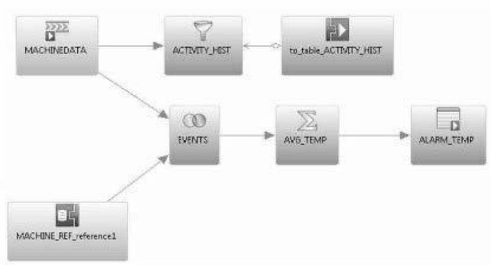


Figure 284: SAP HANA Smart Data Streaming - Design Time

You can define streaming "projects", that define the input streams, continuous queries, and outputs. These projects are defined in CCL – the stream processing language used. A streaming plug-in for the SAP HANA studio provides both a CCL editor and visual editor, along with the testing tools.

SAP HANA Smart Data Integration (SDI)

- High speed event stream capture into SAP HANA tables
- Update SAP HANA tables based on analysis of incoming event streams
  - Streaming projects can apply inserts, updates, and deletes to SAP HANA tables (not limited to event logging)
  - This allows SAP HANA tables to "mirror" event windows in streaming projects
- Join event streams to SAP HANA tables/views
  - Reference HANA tables/views directly in CCL projects
- Pre-load reference data from SAP HANA DB into streaming projects
  - With ability to set refresh interval
- Run event-driven analytics on SAP HANA DB

Figure 285: Streaming Projects Connecting to SAP HANA Tables

You can capture the high speed event stream's data into SAP HANA tables. Update of these tables is based on analysis of incoming event streams. The streaming projects you defined in the previous graphic can apply inserts, updates, and deletes to the SAP HANA tables, and are not just limited to event logging. This allows HANA tables to "mirror" event windows in streaming projects.

- Streaming capacity scales independently of SAP HANA core
- Add streaming nodes for additional stream processing capacity
- Streaming projects connect to SAP HANA database via SAP HANA ODBC driver
- Streaming projects can send events to XS application using http output adapter

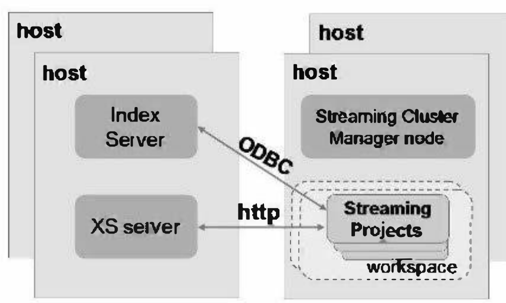


Figure 286: SAP HANA System with One or More Streaming Nodes

You can build an SAP HANA system with a one or more streaming nodes.

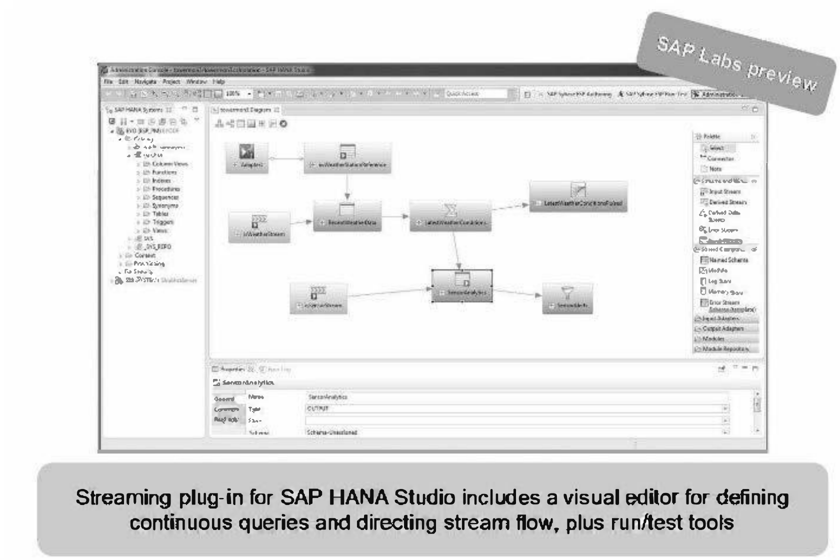


Figure 287: Design-Time Tools in SAP HANA Studio

Design-time tools for ESP are now built into the SAP HANA Studio. We've been working to more tightly integrate ESP with SAP HANA, and we now offer all of the ESP Studio capabilities as a plug-in to SAP HANA Studio. This means that right from within SAP HANA Studio, users can now build ESP projects, run them and test them.

Included:

- Message bus: JMS, IBM MQ, TIBCO
- Web Service SOAP, REST
- http post
- Websockets
- Databases
- Files
- TCP Sockets
- SAP RFC
- SAP Replication Server (in)
- Logfile (in)
- Microsoft Excel (out)
- Email (out)
- HTTP snapshot query (out)

Parsing/Formatting:

- JSON
- XML events
- XML docs
- CSV
- FIX
- JMS Object Arrays

Extensibility:

- Adapter Toolkit supports pluggable transport/parse/format modules written in Java
- API's for C/C++, Java, .NET

Figure 288: SAP HANA Smart Data Streaming -Range of Connectivity Options

This is the range of connectivity that is currently supported. We are constantly adding to the list. Also, this list only represents the off-the-shelf connectivity. It's also easy to implement custom adapters.

Lesson Summary

You should now be able to:

- Understand SAP HANA smart data streaming
- Know where to use Smart Data Streaming
- Understand the value of Smart Data Streaming in certain use cases

Unit Summary

You should now be able to:

- Understand SAP HANA smart data streaming
- Know where to use Smart Data Streaming
- Understand the value of Smart Data Streaming in certain use cases

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Unit 7

SAP HANA Direct Extractor Connection (DXC)

Unit Overview

This Unit will gives you an overview about the functionality of DXC

Unit Objectives

After completing this unit, you will be able to:

- Explain an additional data acquisition technique for working with data from SAP Business Suite systems that has been added to the existing techniques for HANA data acquisition.
- Explain the SAP Business Content DataSource Extractors
- Explain the SAP HANA Direct Extractor Connection in detail
- Explain the Comparison with other SAP HANA Data Acquisition Techniques
- Explain SAP HANA Direct Extractor Connection Setup and Configuration
- know the important DXC Note Numbers
- Explain the DXC Sidecar Variation

Unit Contents

Lesson: Introduction to SAP HANA Direct Extractor Connection Overview

Lesson Overview

This lesson gives a brief overview of SAP Direct Extractor Connection (DXC), which is another data acquisition method. SAP HANA. DXC is a batch-driven data acquisition technique which allows ETL from BW Extractors.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain an additional data acquisition technique for working with data from SAP Business Suite systems that has been added to the existing techniques for HANA data acquisition.

Business Example

You are working for an Organization where they want to connect to an exisitng SAP ERP system to an SAP HANA (peer-to-peer) system and extract SAP Business Content DataSources or customer-specific DataSources to the SAP HANA system.

SAP HANA Direct Extractor Connect (DXC) is a means for providing out-of-the-box foundational data models to SAP HANA, which are based on SAP Business Suite entities.

DXC is also a data acquisition method. The rationale for DXC is essentially simple, low TCO data acquisition for SAP HANA leveraging existing delivered data models.

Customer projects may face significant complexity in modeling entities in SAP Business Suite systems.

In many cases, data from different areas in SAP Business Suite systems requires application logic to appropriately represent business documents.

SAP Business Content DataSource Extractors have been available for many years as a basis for data modeling and data acquisition for SAP Business Warehouse; now with DXC, these SAP Business Content DataSource Extractors are available to deliver data directly to SAP HANA.

DXC is a batch-driven data acquisition technique; it should be considered as a form of extraction, transformation and load although its transformation capabilities are limited to user exit for extraction.

Overview of the DXC Rationale:

- Leverage pre-existing foundational data models of SAP Business Suite entities for use in SAP HANA data mart scenarios:
- Significantly reduces complexity of data modeling tasks in SAP HANA
- Speeds up timelines for SAP HANA implementation projects
- Provide semantically rich data from SAP Business Suite to SAP HANA
- Application logic to give the data the appropriate contextual meaning is already built into many extractors
- Re-uses existing proprietary extraction, transformation, and load mechanism built into SAP Business Suite systems over a simple http(s) connection to SAP HANA
- No additional server or application needed in system landscape
- Provides a mechanism to properly handle data from all delta processing types

- **Challenges**
  - **Data stored in many various tables**, **with high complexity** in many modules of SAP Business Suite systems
  - LT "real-time" approach → Uses base tables in the SAP Business Suite as a basis for data modeling of SAP Business Suite entities

    Project solution: Model SAP entities from scratch
  - In some cases → **Big challenges** b/c of **high complexity** in the SAP Business Suite system
- **DXC Benefits:**
  - Leverage SAP Delivered Business **Content DataSources** → Which are **existing** foundational data models for key entities in SAP Business Suite systems
  - Significantly reduces complexity of data modeling tasks in SAP HANA
  - Speeds up timelines for customer's implementation projects

Figure 289: Pre-Existing Foundational Data Models of SAP Entities for use in SAP HANA

- **Challenges**
  - In many modules of SAP Business Suite systems → Application logic needed to have **semantically rich data** (data appropriately reflecting the state of business documents)
  - LT "real-time" approach → Uses base tables in the SAP Business Suite as a basis for data modeling → Semantically rich data not provided "out of the box"
  - **Project solution: Implement business logic from scratch to properly represent SAP Business Suite data**
  - **It can be extremely challenging to determine proper application logic to implement on a project basis in SAP HANA (depending on use case)**
- **DXC Benefits**
  - DXC → uses SAP DataSource Extractors → provides semantically rich data "out of the box"
  - Ensures the data appropriately represents the state of business documents from ERP
  - **Application logic to "make sense of the data" already built into many extractors**
  - Avoid potentially difficult work of "reinventing the wheel" on a project basis in HANA → reimplement application logic in HANA which is already provided in DataSource extractors

Figure 290: Pre-Existing Foundational Data Models of SAP Entities for use in SAP HANA

- **Challenges**
  - Many SAP Business Content DataSources offer delta processing, aka change data capture
  - Extraction only sends data created, changed, or deleted since the last extraction run → Efficiency
  - Some DataSource extractor types deliver data with special properties → It should not be simply loaded directly into a table in HANA → **This would cause incorrect results in reports**
  - Delta processing types *AIM, AIE, AIED, AIMD, ADD, ADDD* and *CUBE* require data to be loaded into a DataStore Object (DSO) → Activation processing is important for data correctness
  - Standard DSOs in BW include an activation mechanism, which handles the special properties of this data appropriately (e.g. after image only, overwrite, deletion flag, etc.)
  - Without DXC, HANA standalone cannot properly handle data from the aforementioned types
- **DXC Benefits**
  - DXC provides a special In-Memory DataStore Object (IMDSO) for use in HANA standalone
  - **IMDSO Includes the same activation mechanism features as BW DSOs**
  - IMDSO properly handles special requirements of extractor types such as AIM and AIMD, which need DSO activation → ensures correct data in reports
  - IMDSO ensures proper sequencing, overwrite, deletion of data, etc.

Figure 291: Pre-Existing Foundational Data Models of SAP Entities for use in SAP HANA

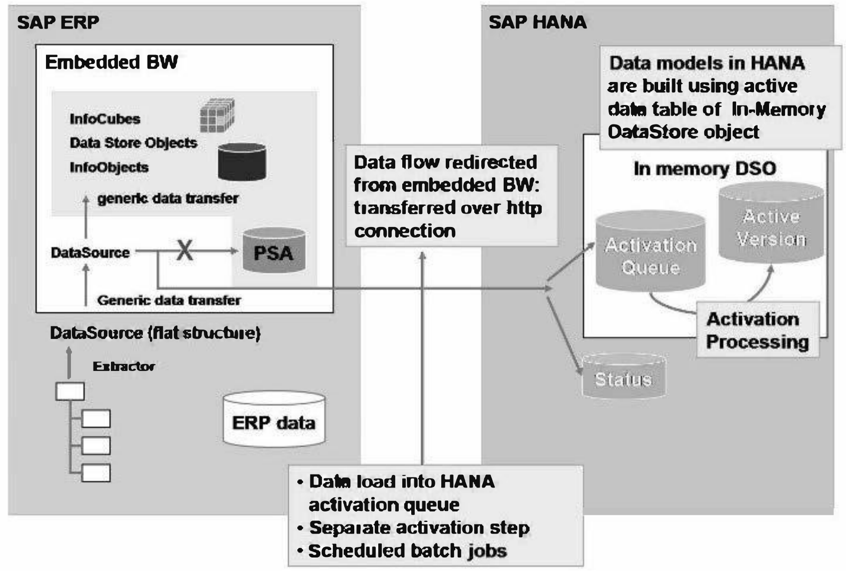


Figure 292: Overview - SAP HANA DXC Concept: Illustration

An SAP Business Suite system is based on SAP NetWeaver. As of SAP NetWeaver version 7.0, SAP Business Warehouse (BW) is part of SAP NetWeaver itself, which means a BW system exists inside SAP Business Suite systems such as ERP (ECC 6.0 or higher). This BW system is referred to as an **"embedded BW system"**.

Typically, this embedded BW system inside SAP Business Suite systems is actually not utilized, since most customers who run BW have it installed on a separate server, and they rely on that one. With the default DXC configuration, we utilize the scheduling and monitoring features of this embedded BW system, but do not utilize its other aspects such as data warehousing or reporting. DXC extraction processing essentially bypasses the normal dataflow, and instead sends data to SAP HANA. The following illustration depicts the default configuration of DXC.

An SAP HANA Optimized DSO is generated in SAP HANA, which directly corresponds to the structure of the DataSource you are working with. This SAP HANA Optimized DSO consists of several tables and an activation mechanism. The active data table of the SAP HANA Optimized DSO can be utilized as a basis for building data models in SAP HANA (attribute views, analytical views, and calculation views).

Data is transferred from the source SAP Business Suite system using an HTTP connection. Generally, the extraction and load process is virtually the same as when extracting and loading SAP Business Warehouse – you rely on InfoPackage scheduling, the data load monitor, process chains, etc. – which are all well known from operating SAP Business Warehouse.

DXC does not require BW on SAP HANA. Also with DXC, data is not loaded into the embedded BW system. Instead, data is redirected into SAP HANA.

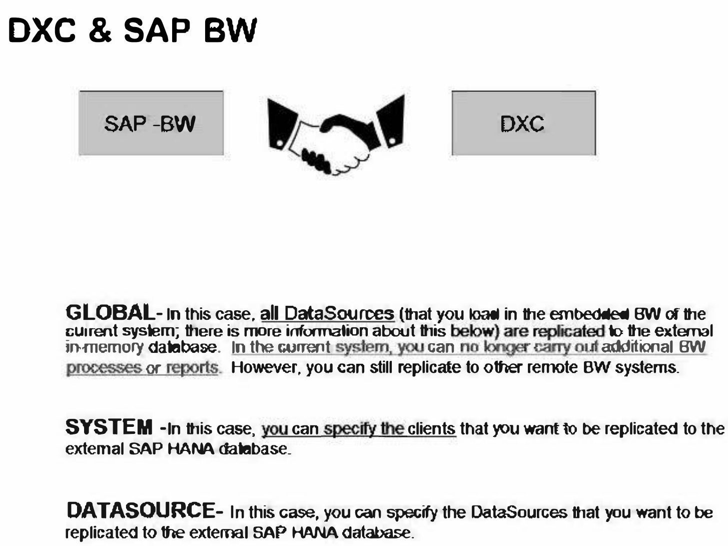


Figure 293: DXC and SAP BW works hand in hand? Sometimes

The details how you can customized these different kinds of "cooperation" between DXC and SAP BW are discribes in SAP Note **1583403** Direct extractor connection to SAP HANA.

Lesson Summary

You should now be able to:

- Explain an additional data acquisition technique for working with data from SAP Business Suite systems that has been added to the existing techniques for HANA data acquisition.

Lesson: SAP Business Content DataSource Extractors

Lesson Overview

This lesson gives an overview to SAP Business Content Extractors

- SAP Busincss Content DataSourcc Extractors

Lesson Objectives

After completing this lesson, you will be able to:

- Explain thc SAP Busincss Contcnt DataSourcc Extractors

Business Example

You want to understand the scope of existing SAP delivered pre-defined extractors to savc timc

- Proprietary Extraction Technology
  - Application based change data capture (aka delta capabilities)
  - Extractors are application based and take data from the context of the application itself
  - Extraction is event/process driven and (in some cases) is accomplished with publishing new or changed data in ERP based Delta Queues for receiving systems (e.g. BW or HANA)
  - Extract structures can easily be enhanced using append structures
  - Transformations can be implemented at the time of extraction using Business Add Ins (BADIs)
  - Extract Structures based on entities in the Business Suite
  - Asynchronous, mass data capable extraction

Figure 294: SAP Business Content DataSource Extractors - Overview

	Transactional data	Master data	Texts	Hierarchies	Totals
ERP & R/3	900	700	2300	100	4000
CRM	160	280	700	40	1180
SRM	30	30	60	10	130
Others	360	210	330	30	930
GRC	20	30	120	10	180
	1470	1250	3510	190	6420

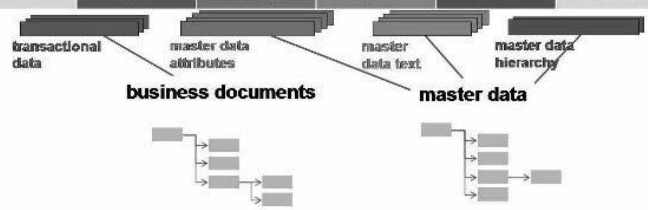


Figure 295: SAP Business Content DataSource Extractors - Thousands of SAP Business Content DataSources Exist

Lesson Summary

You should now be able to:

- Explain the SAP Business Content DataSource Extractors

Lesson: SAP HANA Direct Extractor Details

Lesson Overview

This lesson covers the technical details and pre-requisites

- SAP HANA Direct Extractor Connection Details

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the SAP HANA Direct Extractor Connection in detail

Business Example

You want to implement DXC as the method for Data Provisioning and you want to understand the technical details and the limitations.

- In typical business suite systems, the embedded BW is not utilized
  - Customers typically have separate BW systems
- DXC uses the embedded BW system to enable extraction and monitoring only
  - Data flow is redirected → It gets sent to HANA
  - Note: Modeling in the embedded BW is not part of the DXC solution
- Note: An architectural variation available, which uses a "sidecar" BW instead of the embedded one. See appendix for details.

Figure 296: SAP HANA Direct Extractor Connection - Details

- The extraction from the SAP Business Suite system -> controlled from the Data Warehousing workbench inside the embedded BW
- When data is extracted from, the SAP Business Suite system, it is not loaded into the PSA of the embedded BW instead it is redirected and sent to HANA
  - It gets loaded into in-memory DSO's activation queue
  - Then activated into the active table of the in-memory
- However, in the data load monitor of the embedded BW, the data load into the activation queue in the DSO in HANA appears like data is loading into the PSA in the embedded BW

Figure 297: SAP HANA Direct Extractor Connection - Details

- Delta processing (aka "change data capture")
- Works the same for DXC as it would if BW were the receiving system
  - If the DataSource is delta enabled, then delta-enabled data is available with SAP HANA Direct Extractor Connection
- DXC internally in HANA uses the ICM (Internet Connectivity Manager) – receives XML packages over the http(s) connection
- Mechanism written on the XS Engine (special component for HANA)
  - Receives data packets from ICM, converts format
  - Inserts the records into the activation queue of the in-memory DSO
  - Activation processing → records are go into the active table in proper sequence
- Both the ICM and XS Engine components must be installed in SAP HANA to utilize DXC

Figure 298: SAP HANA Direct Extractor Connection - Details

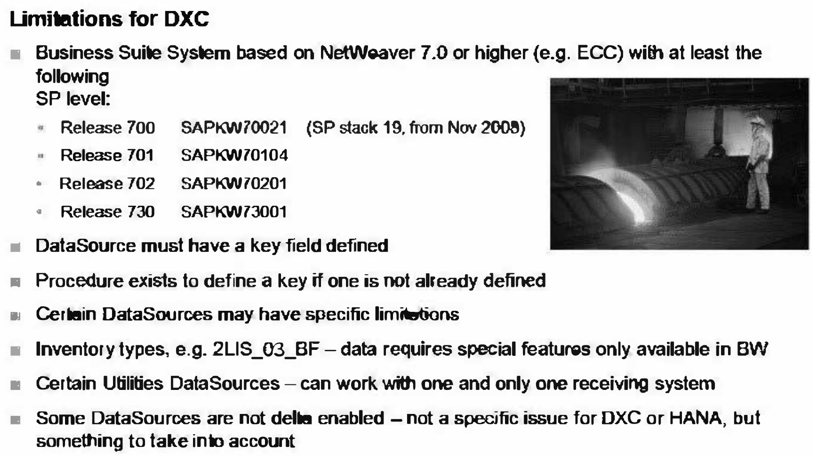


Figure 299: SAP HANA Direct Extractor Connection - Details

- DataSource must have a key field defined
  - Procedure exists to define a key if one is not already defined
- Certain DataSources may have specific limitations
  - Inventory types, e.g. 2LIS_03_BF – data requires special features only available in BW
  - Certain Utilities DataSources – can work with one and only one receiving system
  - Some DataSources are not delta enabled – not a specific issue for DXC or HANA, but something to take into account

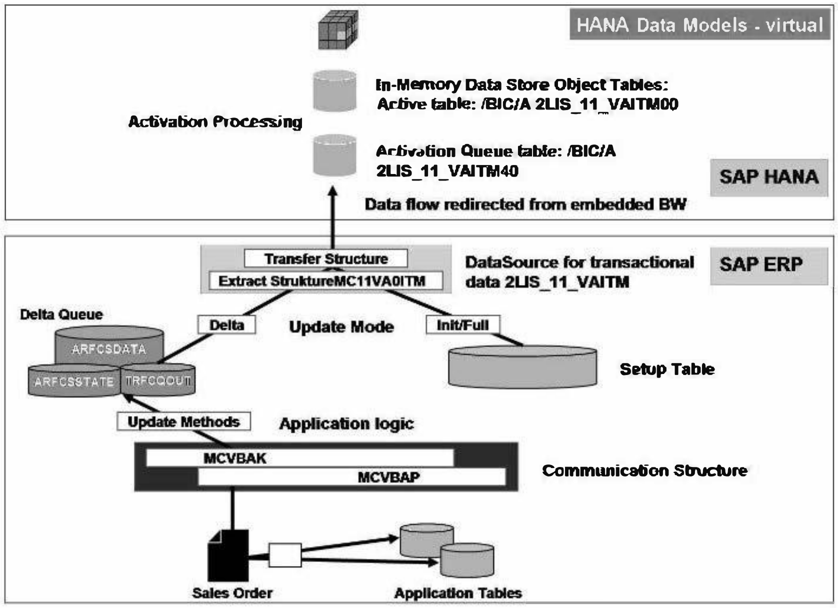


Figure 300: SAP HANA Direct Extractor Connection - SAP Business Suite - DataSource Extractors Example from Sales Order Item Content

The above figure shows an example of a Sales Order Extractor. DXC can use the same delta mechanism. Instead of sending data to BW, it is redirected to SAP HANA.

Lesson Summary

You should now be able to:

- Explain the SAP HANA Direct Extractor Connection in detail

Lesson: Comparison with other SAP HANA Data Acquisition Techniques

Lesson Overview

This lesson compares the different data acquisition

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the Comparison with other SAP HANA Data Acquisition Techniques

Business Example

As an SAP HANA consultant, you are required to give your recommendation of the best data acquisition method.

Direct Extractor Connection	SAP Data Services
ETL type: Simple and straightforward ETL approach; no "premium" features	**ETL type:** Sophisticated ETL tool with extensive valuable features (data quality, metadata mgmt, transformations, etc.)
SAP DataSources: available for all SAP Business Content DataSources (Extractors) and Generic DataSources with a defined key; key can be defined if missing	**SAP DataSources:** available for SAP Business Content DataSources (Extractors), limited to the subset of DataSources released to Operational Data Provider
Support Package Requirement: SP required in the source SAP Business Suite system (ECC 6.0 or higher). DXC is implemented by applying a special SAP note	**Support Package Requirement:** SP must be applied to the source SAP Business Suite system – see SAP note 1522554
Delta handling (change data capture): Yes, for all SAP Business Content DataSources and all delta processing types; uses an SAP HANA Optimized DSO with activation processing	**Delta handling (change data capture):** Yes, except for SAP Business Content DataSources with delta processing types AIM, AIE, AIED, AIMD, ADD, ADDD, CUBE
Software: Uses existing components in SAP HANA (XS Engine,); configuration file imported into SAP HANA	**Software Requirement:** BI 4 Platform 4 and SAP DataServices required
Transformations: very limited - BADI (ABAP) in extraction exit available. When extensive transformations are required, it's recommended to use DataServices	**Transformations:** extensive transformation capabilities available in the DataServices ETL tool

Figure 301: Comparison with other SAP HANA Data Acquisition Techniques Contrast with Data Services

Direct Extractor Connection	SAP Landscape Transformation
Type: Batch-driven ETL. Data comes from SAP delivered Business Content DataSource extractors	**Type:** Trigger-based table replication. Data comes from base tables of SAP Business Suite systems
Real-Time: No. Once every 15 minutes approx. theoretical maximum (depends on DataSource)	**Real-Time:** Yes. Expected lag time typically less than 3 seconds
Delivered Foundational Data Models for SAP Entities: Yes	**Delivered Foundational Data Models for SAP Entities:** Generally no; some RDS content available
Semantically Rich Data: Yes, via SAP delivered Business Content DataSource extractors	**Semantically Rich Data:** No. Semantics must be implemented on a project basis in SAP HANA
System Landscape: Nothing added, uses existing components in SAP HANA	**System Landscape:** SAP NetWeaver 7.01 or higher required
Project Acceleration for Data Marts in HANA: Significant benefit	**Project Acceleration Building Data Marts in HANA:** Available with some RDS packages
Transformations: Limited. BADI in extraction exit available	**Transformations:** Several functions available for data transformations.

Figure 302: Comparison with other SAP HANA Data Acquisition Techniques - Contrast with SLT

Supported Capability Matrix – Part 1 – Data from Tables

	Capability	Long File Replication* Documentation	SLT Replication Documentation	Data Services 4.0 Documentation	Direct Extractor Connection*
Tables	Minimum ERP Version	ECC 6.0	ERP 4.6c	ERP 4.6c	ECC 6.0
	Unicode/Non-Unicode	Unicode Only	Yes	Yes	Yes
	MDMP	No	Yes	Yes	Yes – via how-to guide
	Transparent Tables	Yes	Yes	Yes	Yes –via generic DataSource
	Cluster and Pool Tables	No	Yes	Yes	Yes –via generic DataSource
	Non-SAP ERP Sources	Yes	No	Yes	No
	Compressed Values DB Table	DB2 (9.1, 9.5, 9.7)	Yes	Yes	Yes –via generic DataSource
	Row Compression DB Table	DB2 9.7 only (no support for 9.1 or 9.5)	Yes	Yes	Yes –via generic DataSource
	SAP ERP DB Support	DB2 LUW/UDB Only	All DBs supported under ERP	All DBs supported under ERP	All DBs supported under ERP
	SAP ERP OS Support	AIX, HP-UNIX, SUSE LINUX 10 & SUSE LINUX 11, SPARC/Solaris, (see PAM for version details)	All OS supported under ERP	All OS supported under ERP	All OS supported under ERP
	Transactional Integrity	Yes	No	No	Yes
	Multi-Source Support	Single Source System	Multi Source System	Multi Source System	Multi Source System

* See SAP Note 1513496 for official release limitations

Figure 303: Comparison with other SAP HANA Data - Acquisition Techniques

Further Information

- In the following you will find a few clarifying comments about further considerations.

Type of Data

- It makes sense to consider the type of data that the DataSource provides, in the context of your use case. The SAP HANA Direct Extractor Connection (DXC) is available for all SAP Business Content DataSources. There are a very small number of DataSources related to inventory data, however, which will pose challenges in working with an SAP HANA appliance, since the SAP HANA appliance does not have a concept like BW does of "non-cumulative key figures". There are special features in BW designed for working with inventory data that are not available in SAP HANA natively. For example, a DataSource like 2LIS_03_BF (Material Movements data) is not well suited for use with DXC since it provides data that essentially requires the special features for inventory that BW provides. In such use cases (inventory data), we recommend working with BW itself instead of the SAP HANA appliance.
- Of course, BW on HANA offers its own set of benefits. For a list of DataSources not supported with DXC, please refer to SAP note 1710236.

Supported Capability Matrix – Part 2 - Extractors

	Capability	Long File Replication* Documentation	SLT Replication Documentation	Data Services 4.0 Documentation	Direct Extractor Connection*
Extractors	Minimum ERP Version			See SAP Note : 1558737	ECC 6.0 or higher
	Unicode/Non-Unicode			Yes	Yes
	MDMP			Partial (Not recommended)	Project solution with how-to-guide
	Transparent Tables			N/A	N/A
	Cluster and Pool Tables			N/A	N/A
	Non-SAP ERP Sources	N/A	N/A	Available	N/A
	SAP ERP DB Support			All DBs supported under ERP	All DBs supported under ERP
	SAP ERP OS Support			All OS supported under ERP	All OS supported under ERP
	Multi-Source Su rt			Multi Sou ce System	Mulliple Source Systems – clients must have unique logical system names
	Semantically Rich Data Definitions	No	No	Yes	Yes

* See SAP Note 1513496 for official release limitations

Figure 304: Comparison with other SAP HANA Data - Acquisition Techniques

Lesson Summary

You should now be able to:

- Explain the Comparison with other SAP HANA Data Acquisition Techniques

Lesson: SAP HANA Direct Extractor Connection Setup and Configuration

Lesson Overview

This lesson covers the technical setup and configuration for SAP Direct Extractor Connection

- SAP HANA Direct Extractor Connection Setup and Configuration

Lesson Objectives

After completing this lesson, you will be able to:

- Explain SAP HANA Direct Extractor Connection Setup and Configuration

Business Example

You want to implement DXC. Therefore you need to find out about the setup, configuration and pre-requisites

How to use DXC - step by step

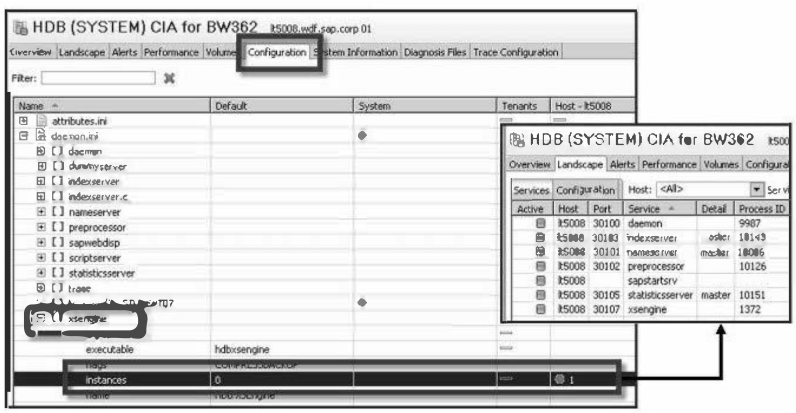


Figure 305: Step 1: Enabling XSEngine and SAP WebDispatcher Service

Ensure that the necessary technology components in SAP HANA are running, that is:

- the XSEngine
- the SAP Web Dispatcher service

If these services are not running, proceed as follows to enable them:

1. Open SAP HANA studio.

2. Right-click your SAP HANA instance and select Administration from the context menu to open the Administration perspective of your SAP HANA instance.

3. Select the Configuration tab. Expand the daemon.ini section and then the sapwebdisp section.

For the instances parameter, a green light showing the number 1 on the Host column should be displayed.

4. If the value 1 is not displayed, double-click the instances parameter and change the configuration value from 0 to 1.

5. Click Save to save the new value.

6. Continuing in the daemon.ini section, expand the xsengine section.

For the xsengine parameter, a green light showing the number 1 on the Host column should be displayed.

7. If the value 1 is not displayed, double-click the instances parameter and change the configuration value from 0 to 1.

8. Click Save to save the new value.

Check if the XSEngine is running by accessing the following URL: http://<hostname>:80<instance number>

The port name in this URL may differ on your system, depending on the instance number of your installation.

If the XSEngine and SAP Web Dispatcher service are operational, you should see the following information on your screen: SAP XSEngine is up and running.

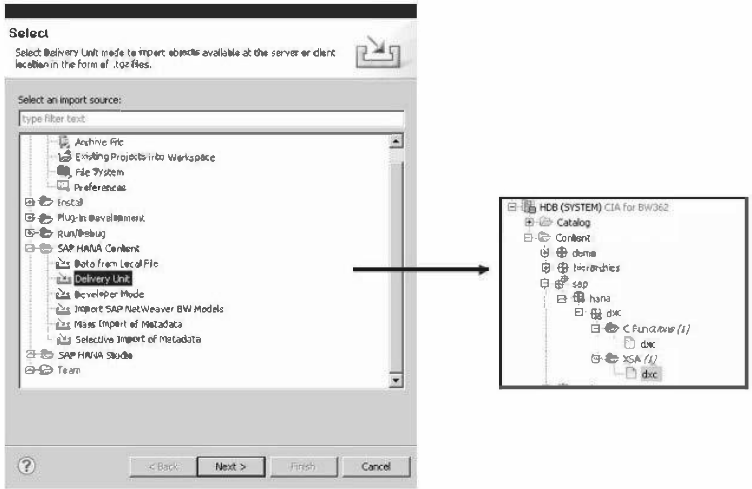


Figure 306: Step 2: Delivery Unit Import

Prerequisites

- Prior to importing the delivery unit, prepare the delivery unit archive for import. Acquire the
- delivery unit archive from the SAP Service Marketplace at
- http://service.sap.com/swdc and save it to a location on your local computer.
- Importing the delivery unit
- For importing the delivery unit, do the following:

1. Open SAP HANA studio and change to the SAP HANA modeler perspective.

2. On the Quick launch tab, in the Content section choose Import.

3. In the Import dialog, expand the HANA Content node, choose Delivery Unit and click Next.

4. Select Client and navigate to the location on your local computer where you have stored the delivery unit archive.

5. Select the delivery unit archive and click OK.

The status in the Object import simulation should display green lights.

6. Keep the existing defaults, and click Finish.

In the lower right part of the screen you should see a tab display for Job Log, with a progress indicator under Current. Once the delivery unit import is completed, the message Completed Successfully is displayed.

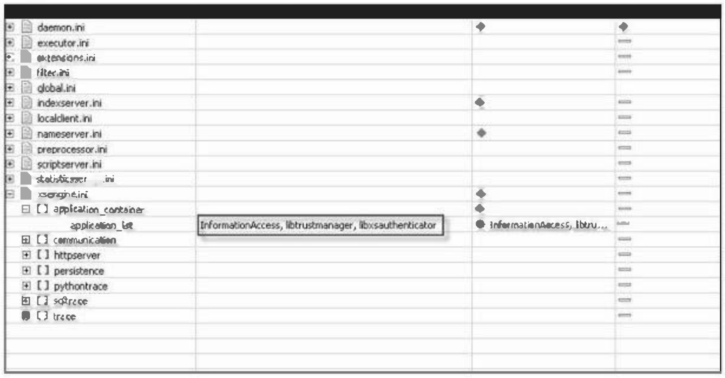


Figure 307: Step 3: Application Server Configuration

The use of the DXC application has to be enabled with the XSEngine component of SAP HANA. To do so, an entry has to be added in the application server configuration.

1. Open the SAP HANA studio and select the instance of the SAP HANA database.
2. Open the Administration perspective and select the Configuration tab.
3. Expand the xsengine.ini section and then the application_container section.
4. Right-click the application_list parameter and select Change from the context menu.
5. In the New Value field, enter libxsdxc.

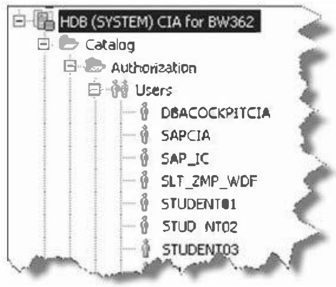
If another value is already there, put a comma after the existing value and add the value libxsdxc, for example searchservice, libxsdxc.

6. Click Save.

- In order to make the new configuration effective, the XSEngine has to be restarted.

1. In the Administration perspective of SAP HANA studio, choose the Landscape tab.
2. Right-click the xsengine service and select Stop. The status light will change to yellow and then to red.
3. Right-click the xsengine service again and select Start missing services.

- Check if DXC is operational. The DXC application should be accessible now by using the following URL: http://<hostname>:80<instance number>/dxc/dxc.xscfunc
- If the check is successful, you will be prompted to save a file with the name dxc.xscfunc to your computer. The contents of this file are not important; the test is successful if calling this URL produces this file.



- Create a user who has the privileges to execute the DXC extraction and load.
- Add the roles PUBLIC and MONITORING

Figure 308: Step 4: Creating a DXC User in SAP HANA

Create a user who has the privileges to execute the DXC extraction and load. To create this user – in the following sections referred to as the DXC user – do the following:

1. Open SAP HANA studio and select your SAP HANA system.
2. In the navigation tree, select Catalog → Authorization → Users.
3. Right-click Users and select New User.
4. In the User Name field, enter an appropriate name, for example DXCUSER.
5. Select internal authentication, enter a password and confirm the password.

- On the Granted Roles tab, add the roles PUBLIC and MONITORING.
- Click the green Deploy icon.

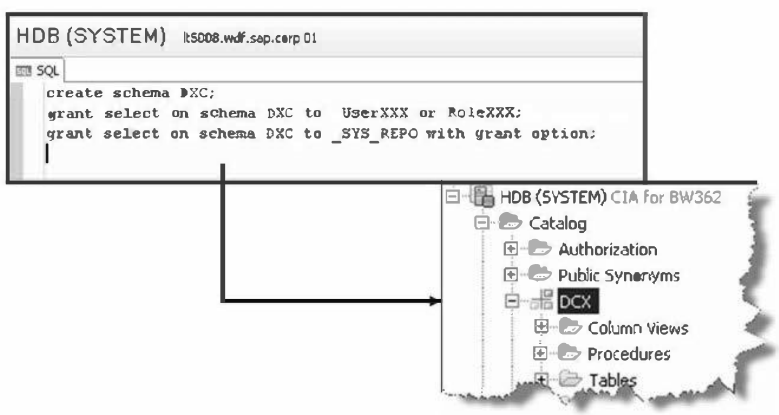


Figure 309: Step 5: Creating a DXC Schema in SAP HANA

Create a schema for use with the SAP HANA Direct Extractor Connection which is owned by the DXC user.

You should create a distinct schema for each specific SAP Business Suite system that you will connect to this SAP HANA system with DXC.

1. In SAP HANA studio, execute the following SQL statement:

create schema <SCHEMANAME> owned by <DXCUSERNAME>

- Example:

  create schema R3TSCHEMA owned by DXCUSER

2. Click Deploy or press F8 to create the schema.

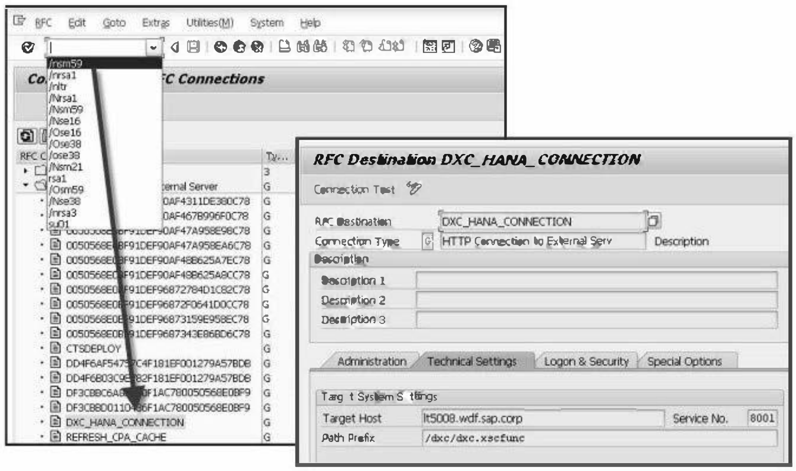


Figure 310: Step 6: Create an HTTP Connection to the SAP HANA System

In the SAP Business Suite system, create an HTTP destination of type G using the transaction SM59.

1. On the Configuration of RFC Connections screen, select the node HTTP Connections to

- External Server and click the Create icon.
- Provide a name for the HTTP destination, for example DXC_HANA_CON-NECTION_<SID>.
- In the Technical Settings tab, enter the target host name and the port number of your remote SAP HANA system in the Target Host and Service No. fields.
- The port number is 80<instance number> of your SAP HANA database.
- In the Path Prefix field, enter /dxc/dxc.xsfunc.
- On the Logon and Security tab, choose Basic Authentication and provide the user name and password of your DXC user.
- Click Save.

Figure 311: Step 7: Configure DXC HTTP Interface Destination and maintain DataSource parameters (I)

Configure DXC HTTP Interface Destination

Create an entry in the table RSADMIN, which is the primary BW configuration table. In this table, the HTTP destination to the SAP HANA system created earlier is designated as the connection used for DXC.

1. Use transaction SA38 to execute the program SAP_RSADMIN_MAINTAIN.

In the OBJECT field, enter the value PSA_TO_HDB_DESTINATION.

In the VALUE field, enter the name of the HTTP destination you created, for example DXC_HANA_CONNECTION_<SID>.

- Make sure that you use upper and lower case letters for the name of the HTTP destination correctly.
- Click Insert, and then click Execute (F8) to create the table entry.
- Choose the System-Wide Setting for DataSources
- In this configuration step, you specify the extent of the use of DXC in the source SAP Business Suite system. Keep in mind that subsequent references to SAP BW typically refer to the embedded BW system which lives inside the SAP Business Suite system, unless you are using the "sidecar" approach discussed in section Appendix – DXC System Landscape Variants: The "Sidecar" Approach.

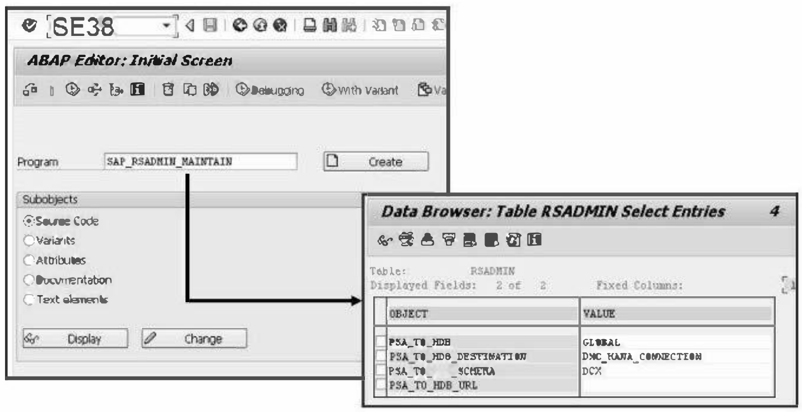


Figure 312: Step 7: Configure DXC HTTP Interface Destination and maintain DataSource parameters (II)

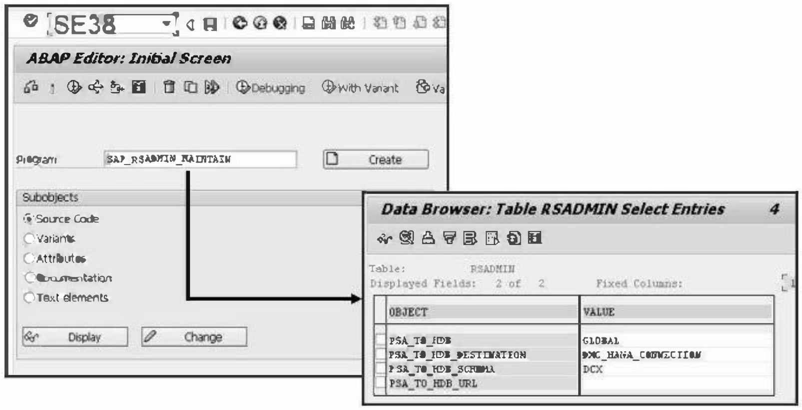


Figure 313: Step 7: Configure DXC HTTP Interface Destination and maintain DataSource parameters (III)

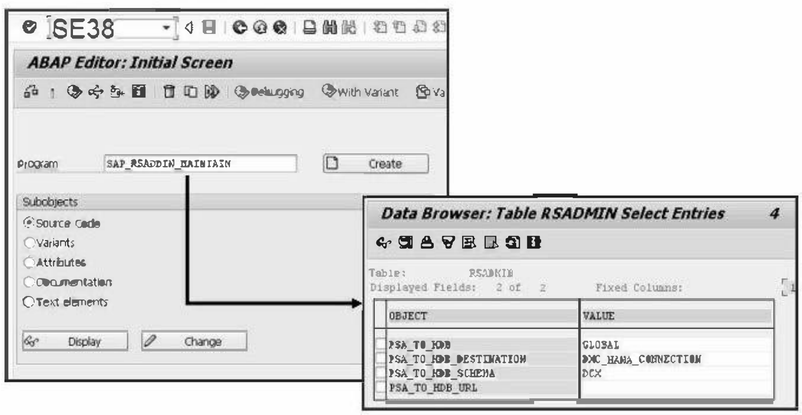


Figure 314: Step 7: Configure DXC HTTP Interface Destination and maintain DataSource parameters (IV)

Replicate DataSources

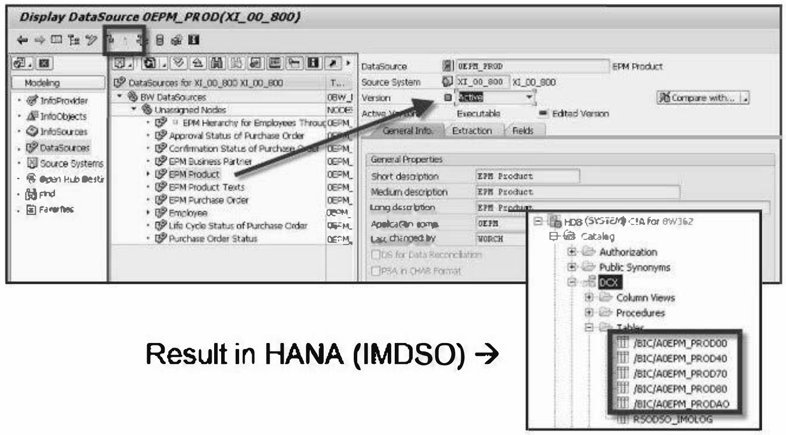


Figure 315: Step 8: Configuration Steps Specific to SAP Business Warehouse (I)

The next steps involve working in the BW Data Warehousing workbench (transaction RSA1). Consider the client to use for the BW client carefully. Be aware that once you have decided which client in your system is the BW client, the transaction RSA1 or other BW-related functions cannot be used in any other client. It is very difficult to change to another client later, as once you have executed transaction RSA1, many configurations are performed automatically.

In order to be able to transfer data, you have to create source systems for any clients in the SAP Business Suite system that should be able to extract data and load it to the SAP HANA database. For further information about creating source systems for this purpose, please refer to the BW documentation at *http://help.sap.com/nw_platform → <your SAP NetWeaver Platform version> → Application Help → SAP Library → <your language> → SAP NetWeaver Business Warehouse.*

1. Right-click the source system you are working with and select Replicate DataSources. If prompted about the type of DataSource, choose DataSource for all; do not choose the old 3.x type DataSource.
All DataSources that have been installed using transaction RSA5 are transferred to the Data Warehousing workbench.

2. Right-click the DataSource(s) you want to work with and select Change.

3. In the Change DataSources dialog, click the Activate icon.

This creates an SAP HANA Optimzed DSO in the SAP HANA database that corresponds to the DataSource structure.

Replicate DataSources

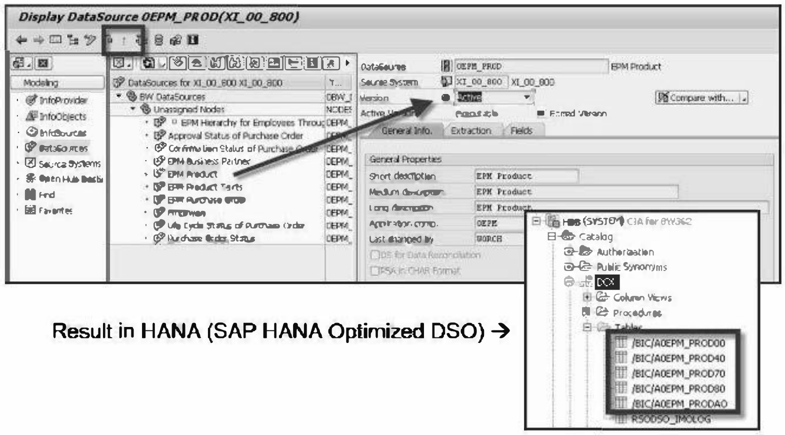


Figure 316: Step 8: Configuration Steps Specific to SAP Business Warehouse (II)

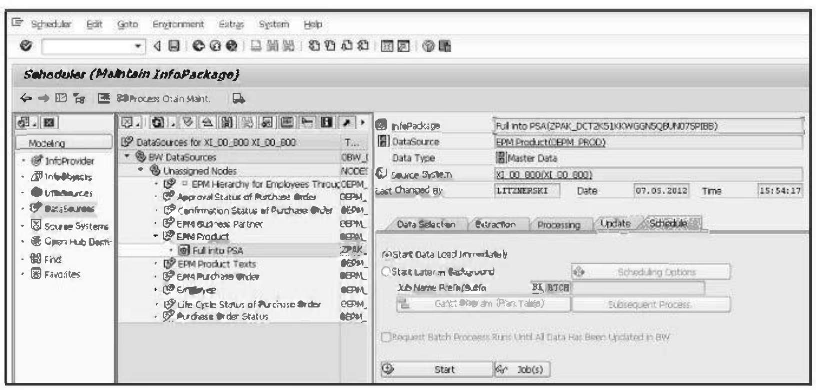


Figure 317: Step 9: Create InfoPackages

In order to be able to load the DataSources into the SAP HANA database, you have to create InfoPackages for the DataSources. In some cases, it is feasible to have delta processing (change data capture). In this case, create the following InfoPackages:

- One InfoPackage for the delta initialization
- One InfoPackage for the regular delta data loads

Otherwise, if delta processing is not available, create an InfoPackage for full load, or several InfoPackages for full load using selection criteria.

After the InfoPackage has been created, schedule it to load data into your SAP HANA Optimised DSO in the SAP HANA database.

1. In the Data Warehousing workbench, right-click the DataSource you are working with and select Create InfoPackage.

2. In the Create InfoPackage dialog, select the appropriate options and click Save.

3. On the Schedule tab, select the appropriate time for the job to execute, and click Start to schedule the extraction job.

- Monitor Data Load in the Source SAP Business Suite System
- Verifying Data Transfer in the SAP HANA Database
- Create a Process Chain for Regular Data Transfer

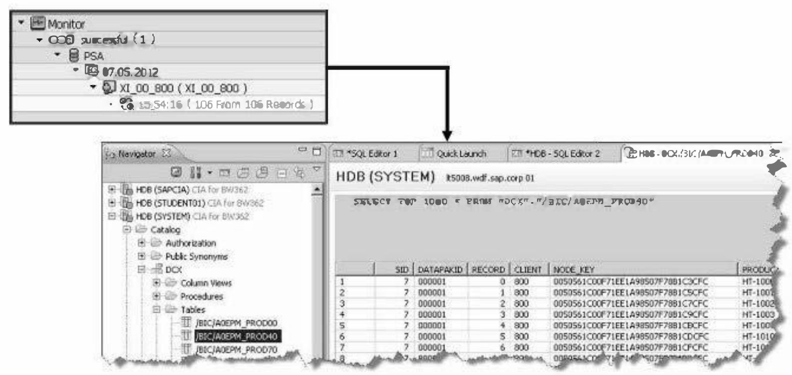


Figure 318: Step 10: Scheduling and Monitoring (I)

After the data load is started, you can monitor the status of the InfoPackages.

1. In the Data Warehousing Workbench, select the InfoPackage you want to monitor and click the Monitor icon.
The Monitor - Administrator Workbench dialog appears.

- 2. Click the Status tab to get detailed information about the data load.

  The data is loaded into the activation queue table for the corresponding SAP HANA Optimized DSO in the SAP HANA database, although a status message is displayed stating that the request was successfully loaded to PSA.

3. Click the Details tab to get detailed information about the records that have been transferred.

- You can navigate through the structure of the processed data packets:

1. Look for Processing (Data Packets), and choose a data package.

2. Expand the node Update PSA. You should see a message like this:

Data package 1 saved to remote SAP HANA DB

- You can also expand the node for Subsequent Processing; errors in the activation process will be displayed here.
- After the data for a given extraction job has successfully been loaded into the activation queue of the SAP HANA Optimized DSO, the data is immediately activated into the active data table of the SAP HANA Optimized DSO.
- The order in which data is loaded is important for data consistency. Therefore, all subsequent data activation for an SAP HANA Optimized DSO will be blocked if a (failed) request is still available in the activation queue table ofthe SAP HANA Optimized DSO. If you encounter such an activation failure, please refer to SAP note 1665553.

- Monitor Data Load in the Source SAP Business Suite System
- Verifying Data Transfer in the SAP HANA Database
- Create a Process Chain for Regular Data Transfer

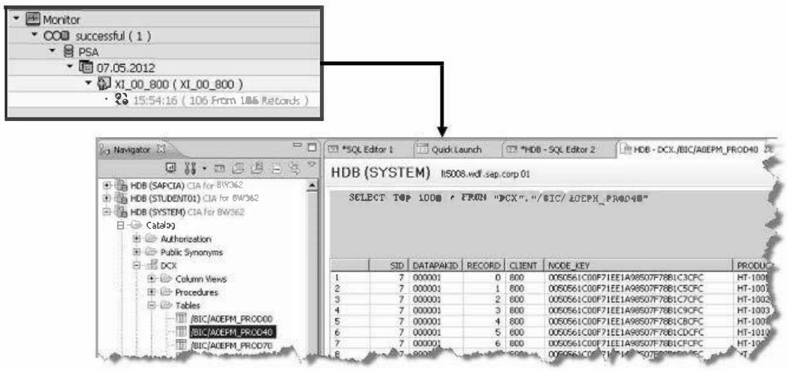


Figure 319: Step 10: Scheduling and Monitoring (II)

1. Log on to SAP HANA studio with the DXC user.
2. Expand the Catalog node, and then expand the SYS node.
3. Locate the M_EXTRACTORS view.

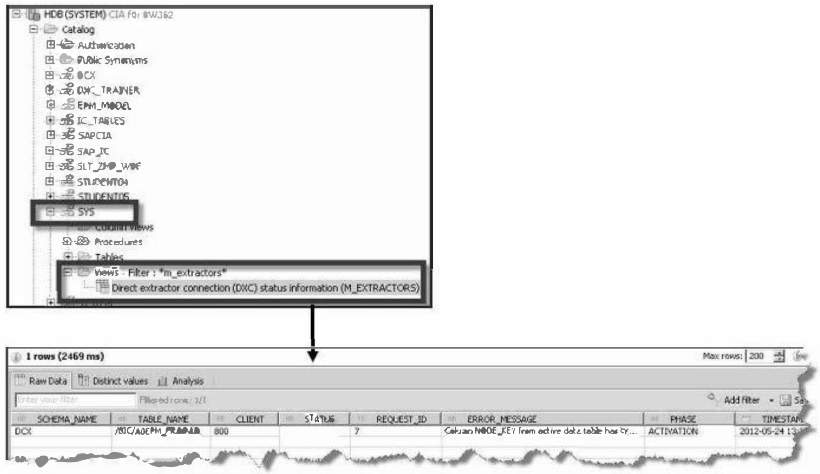


Figure 320: Step 11: Monitoring the Activation Process of SAP HANA Optimized DSO (II)

Exercise 5: Creating an Extractor Connection (DXC)

Exercise Objectives

After completing this exercise, you will be able to:

- Create and use a Direct Extractor Connection (DXC)

Business Example

You are at a customer site and you need some SAP ECC data directly in HANA. You need these data with the least amount of effort.

Task 1:

Create a DXC DataSource and load ECC data directly into HANA.

1. Log on to the TAS system and check the DXC parameters in table RSADMIN using transaction SE16.

Task 2:

Create your own DataSource based on credentials of existing DataSource HA_SFLIGHTXX.

1. Start transaction SBIW and create a generic DataSource (HA_SLFLIGHT_##) for transaction data.
   Application Component → TRAINING
   Description → HA_SFLIGHT
   View/Table → SFLIGHT

   In the popup to create a new Object entry choose *Local* ($TMP) and Save.

   In the Data Source: Customer version Edit screen save again without any changes to the defaults.

Task 3:

Replicate your DataSource based on application component **TRAINING.**

1. Replicate your DataSource based on application component **TRAINING**

Continued on next page

Task 4:

Check the new SAP HANA Optimized DSO directly in the SAP HANA Studio.

1. Check the new SAP HANA Optimized DSO and identify the tables in the SAP HANA Studio.

Task 5:

Create an InfoPackage for loading data

1. Create an Infopackage for your datasource.on *Create InfoPackage*. Choose all default values and start the loading process by clicking on *Start* in the last tab.

   Verify the data in BW (load monitor) and on HANA system (table preview on the active table)

Task 6:

Delete your DataSource in transaction RSA1 and have a look to the coresponding HANA tables. They will be deleted.

1. Right-click on your DataSource in the SAP system and select Delete. Then check it in SAP HANA.

Solution 5: Creating an Extractor Connection (DXC)

Task 1:

Create a DXC DataSource and load ECC data directly into HANA.

1. Log on to the TAS system and check the DXC parameters in table RSADMIN using transaction SE16.

   a) Open the SAP Logon program

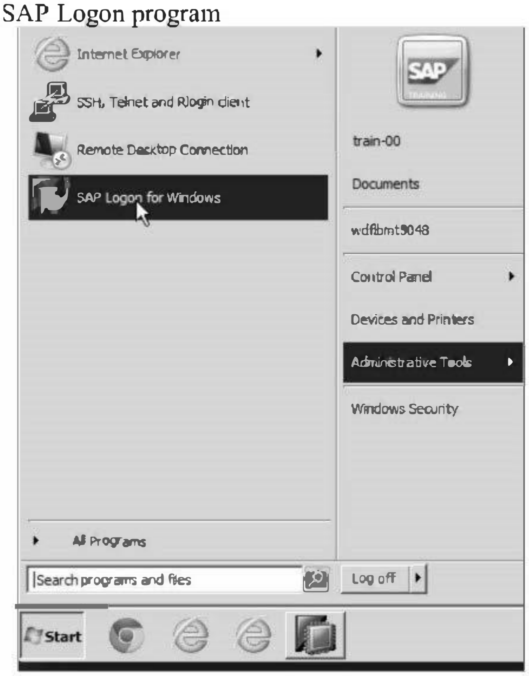


   Figure 321: Open the SAP Logon program

   b) Logon to the TAS system

Continued on next page

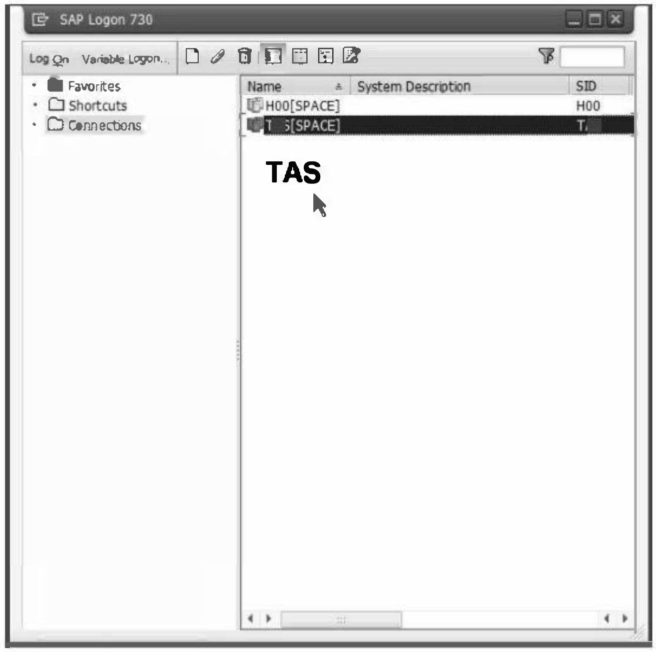


Figure 322: Log on to the TAS system

c) Use the STUDENT-## user to log on (remember the dash, ## is your student number). The password will be INITIAL if you log on the first time. Change it to TRAINING when asked.

Continued on next page

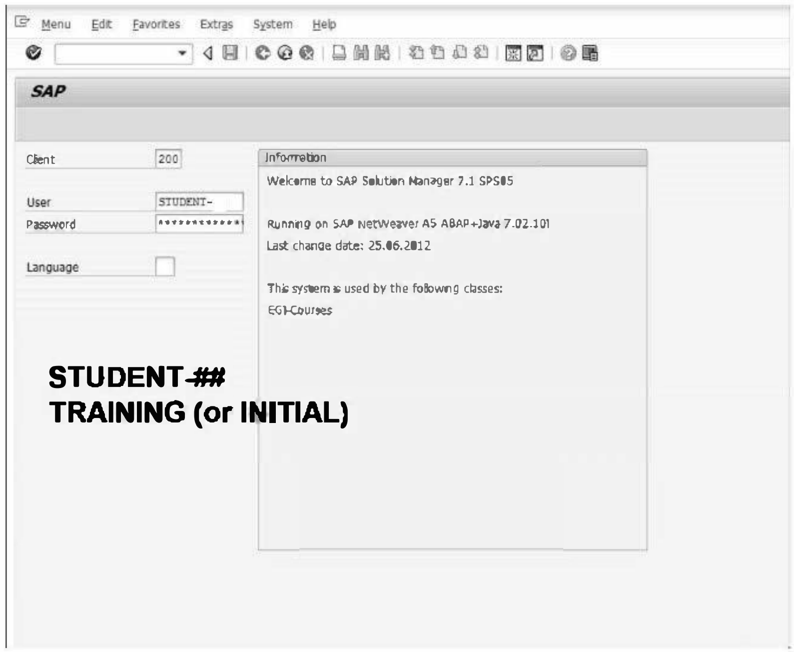


Figure 323: Logon with your STUDENT-## user

d) Start transaction SE16

Continued on next page

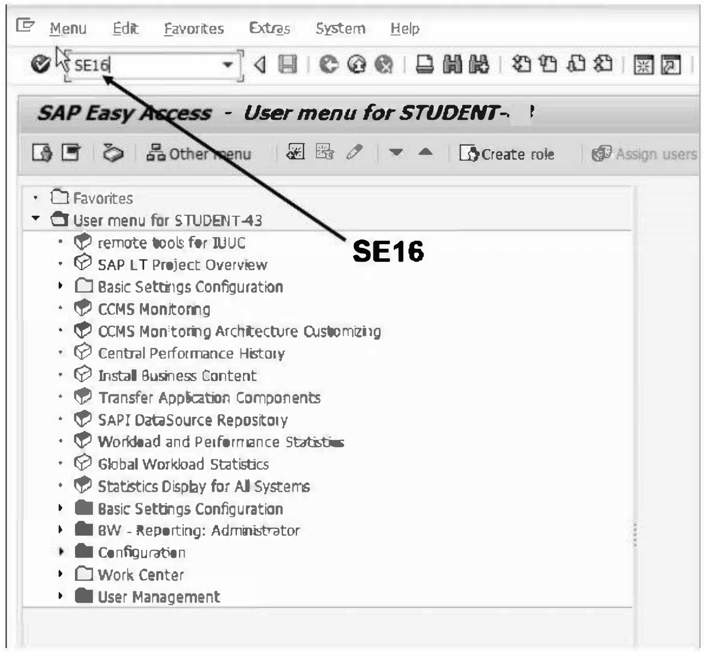


Figure 324: Open transaction SE16

c) Open table RSADMIN

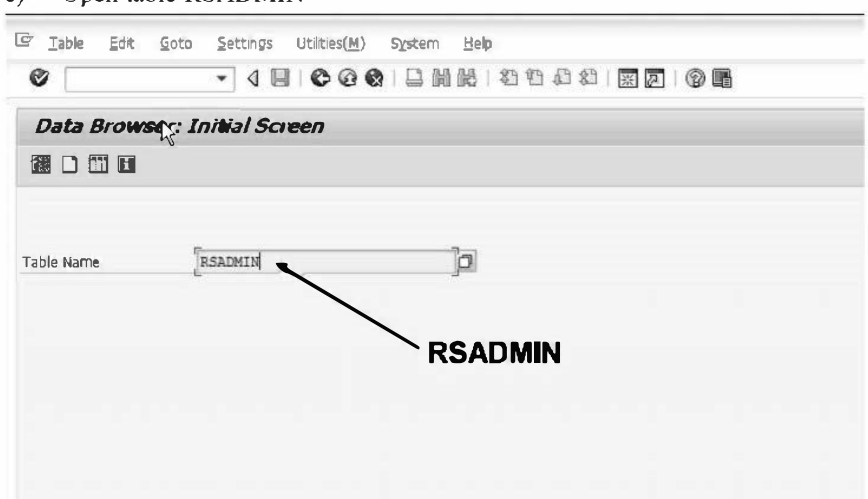


Figure 325: Open table RSADMIN

Continued on next page

f) Check the parameters in table RSADMIN. In this step you are only checking the parameters. Then you know where to find them later when you need them.

. Check the following entries:

PSA_TO_HDB → *"Global"* (All activated DataSources will create an SAP HANA Optimized DSO)

PSA_TO_HDB_DESTINATION → *"DXC_CONNECTION"* (Based on the HTTP Connection prepared in SM59)

PSA_TO_HDB_Schema → *"DXC<LOCATION>"* (Schema for automatically generated SAP HANA Optimized DSO tables based on new DataSources)

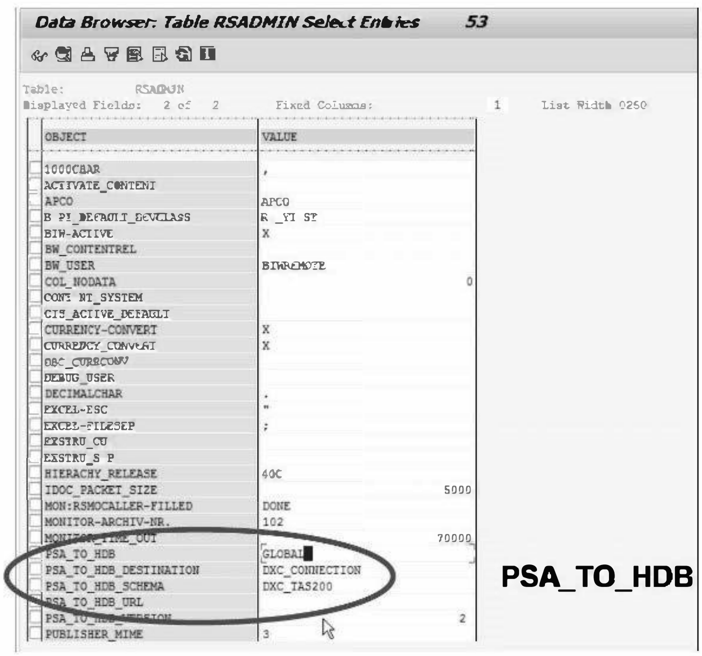


Figure 326: Check the entries

Continued on next page

Task 2:

Create your own DataSource based on credentials of existing DataSource HA_SFLIGHTXX.

1. Start transaction SBIW and create a generic DataSource (HA_SLFLIGHT_##) for transaction data.
   Application Component → TRAINING
   Description → HA_SFLIGHT
   View/Table → SFLIGHT

   In the popup to create a new Object entry choose *Local* ($TMP) and Save.

   In the Data Source: Customer version Edit screen save again without any changes to the defaults.

   a) Type in transaction /nSBIW to open the generic DataSource maintenance in the embedded BW system.

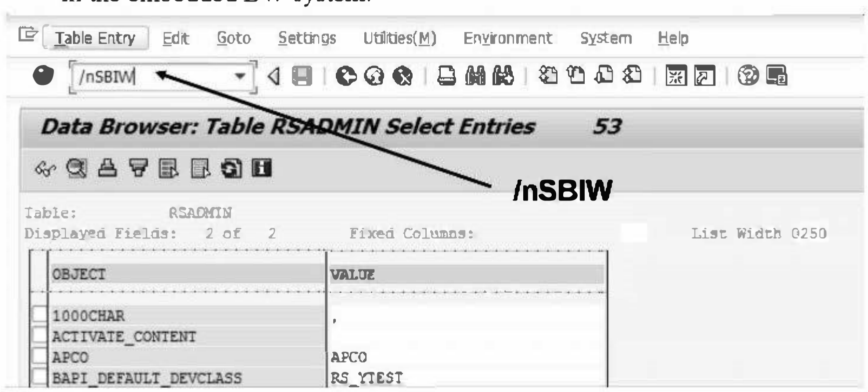


Figure 327: Start Transaction SBIW

   b) Expand node *Generic DataSource* and click on *Maintain Generic DataSources*.

Continued on next page

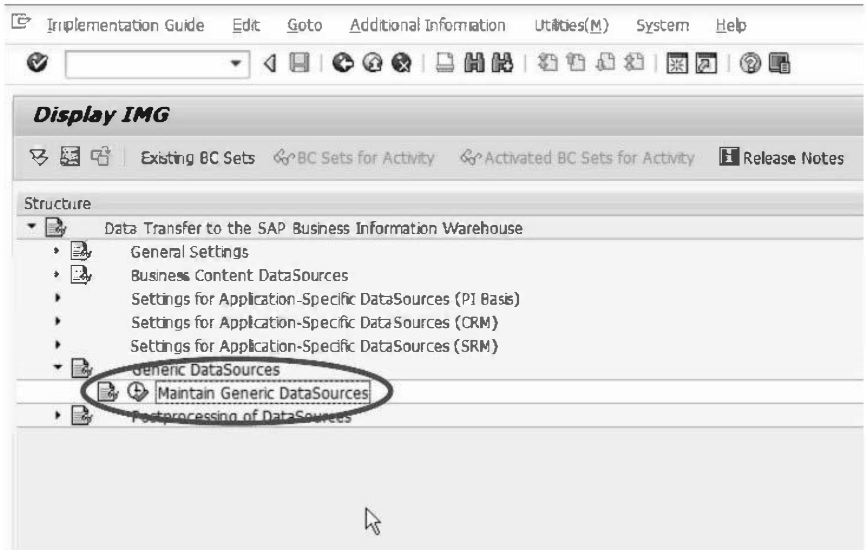
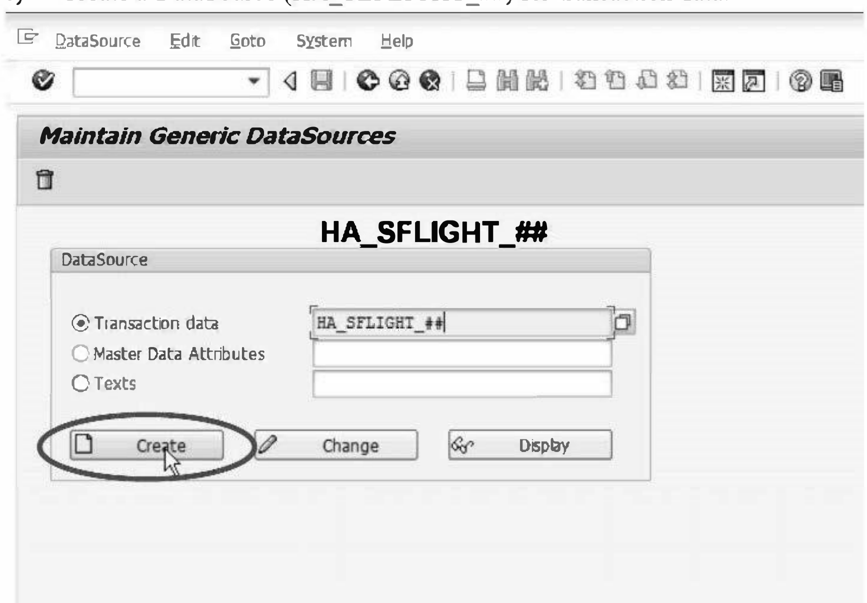


Figure 328: Maintain Generic DataSources

c) Create a DataSource (HA_SLFLIGHT_##) for transaction data.



Continued on next page

Figure 329: Create a DataSource

d) Specify the parameters for the DataSource as follows:
Application Component → *TRAINING*
Description → *HA_SFLIGHT*
View/Table → *SFLIGHT*

Save the DataSource. (Top toolbar, icon)

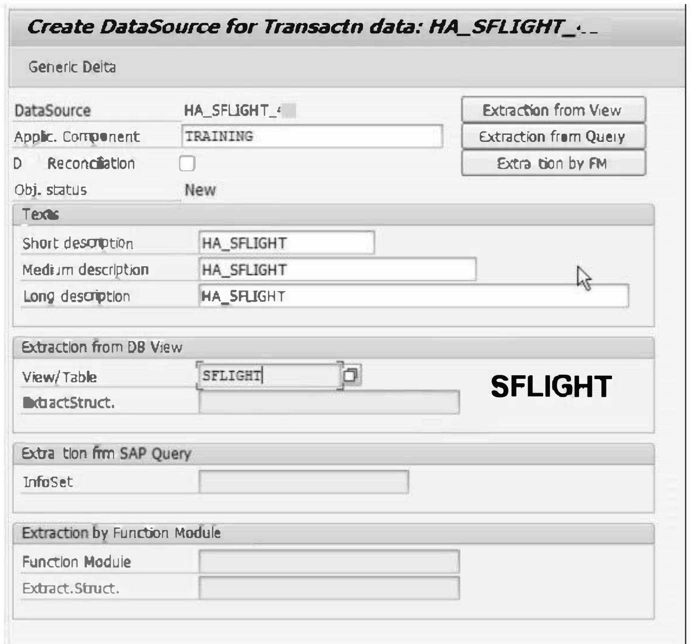


Figure 330: Specify the parameters and save

e) In the popup to create a new Object entry choose *Local* ($TMP) and Save.

Continued on next page

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

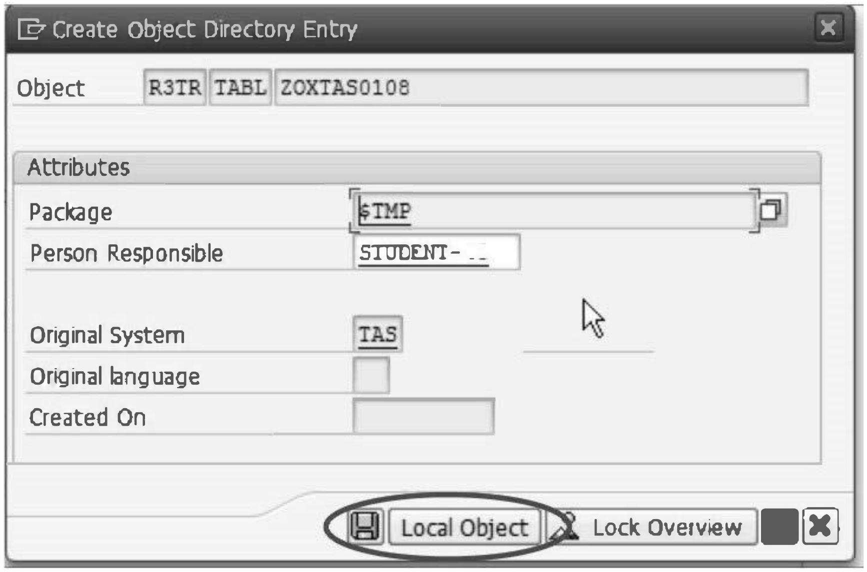


Figure 331: Specify that this is a local object (does not need to be transported to a production system)

f) In the Data Source: Customer version Edit screen save again without any changes to the defaults.

Continued on next page

Figure 332: Save

g) You have now successfully created a data source.

Continued on next page

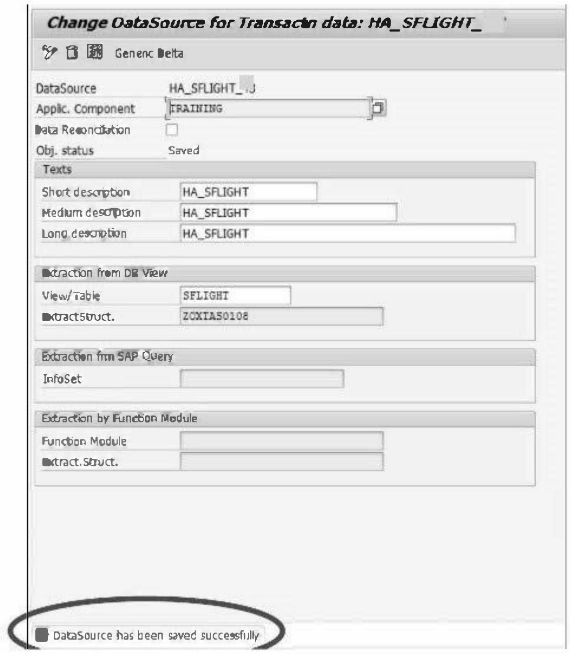


Figure 333: Successfully saved

Task 3:

Replicate your DataSource based on application component **TRAINING.**

1. Replicate your DataSource based on application component **TRAINING**

   a) Start transaction RSA1.

Continued on next page

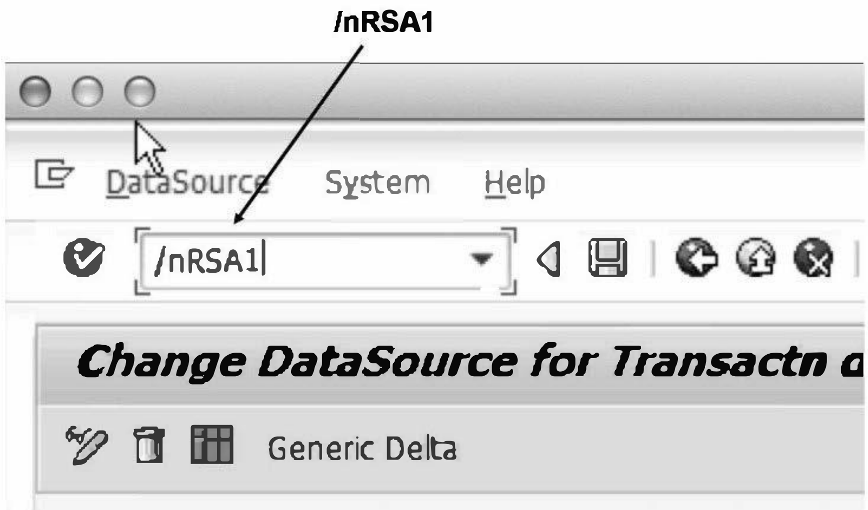


Figure 334: Start transaction RSA1

b) If you get a question on Documentation, you can just answer *No*.

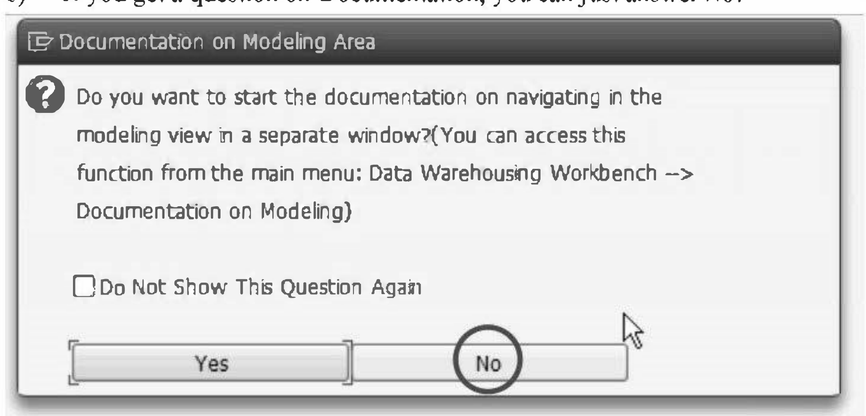


Figure 335: Decline documentation

c) Open Modeling-DataSources and choose the DataSource tree for the **TASCLNT200** system.

Continued on next page

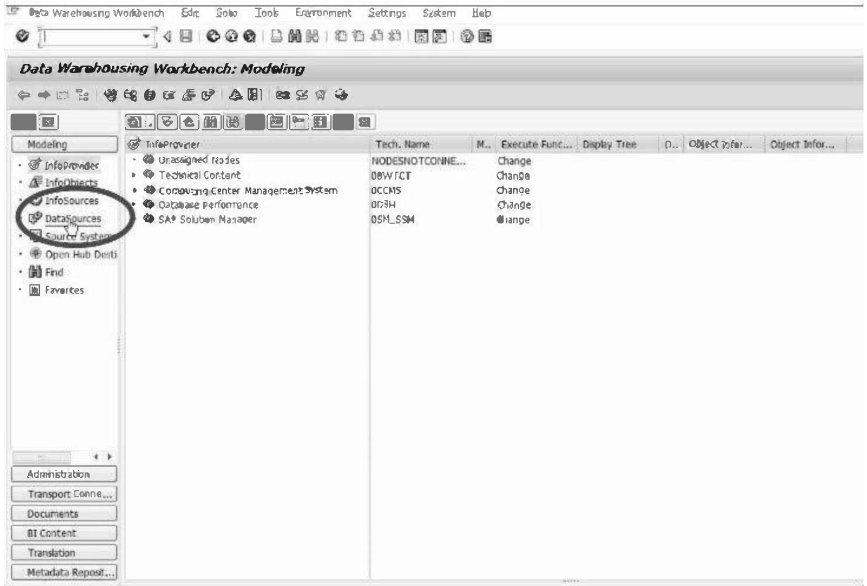


Figure 336: Open the DataSources tree

d) Select TRAINING

Note: PLEASE use application component TRAINING ONLY!!

Continued on next page

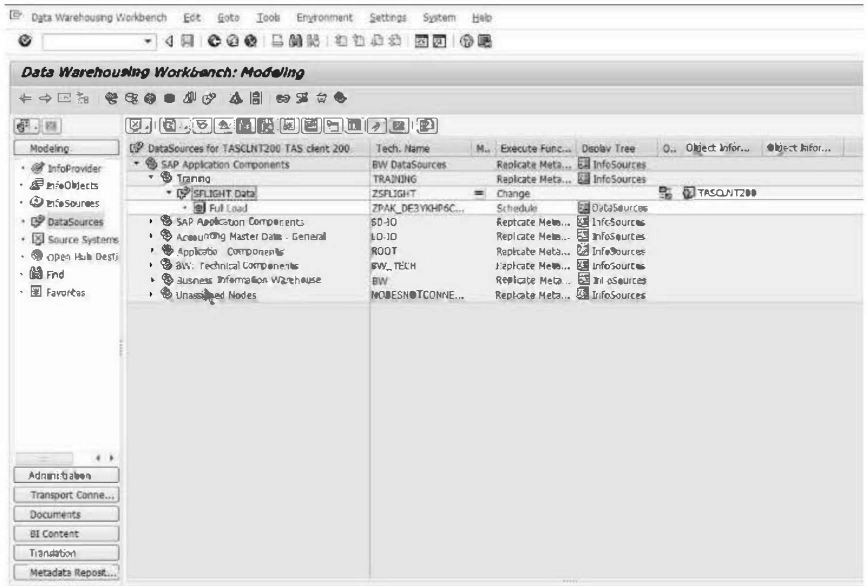


Figure 337: Select the training component

e) Right- click on Training, and select *Replicate Metadata*

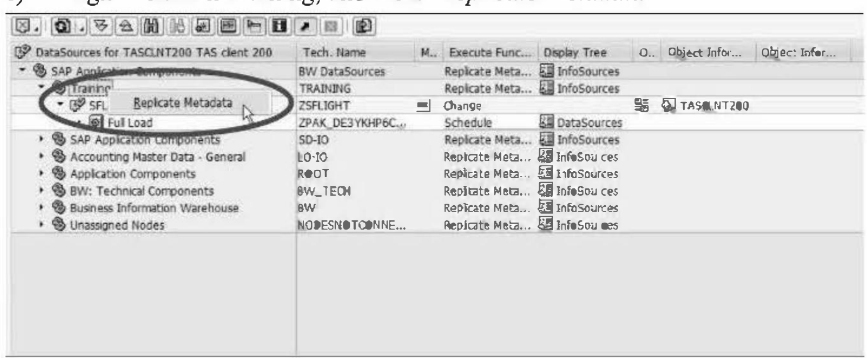


Figure 338: Replicate Metadata

Choose *Replicate Metadata All DataSources* for application component DXC will be replicated.

f) For new DataSource choose type as DataSource (RSDS) and confirm.

Continued on next page

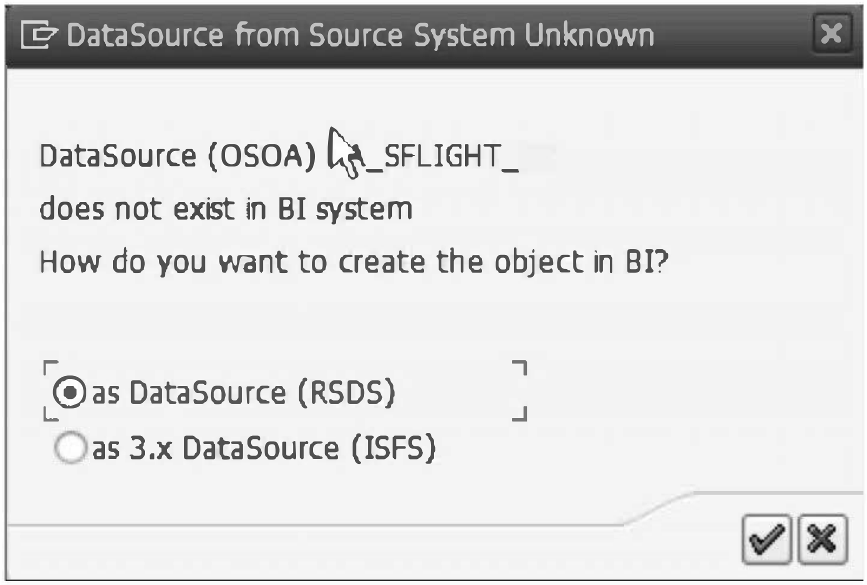


Figure 339: Choose type as DataSource (RSDS)

g) Your DataSource HA_SFLIGHT_## has been added

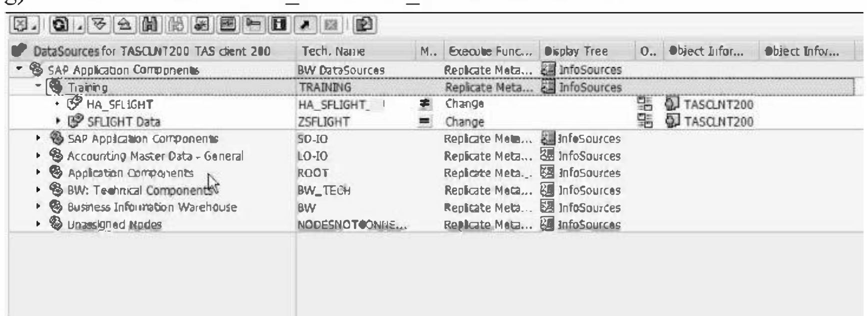


Figure 340: Your DataSource was added

h) Right-Click your DataSource HA_SFLIGHT_## and choose *Change*.

Continued on next page

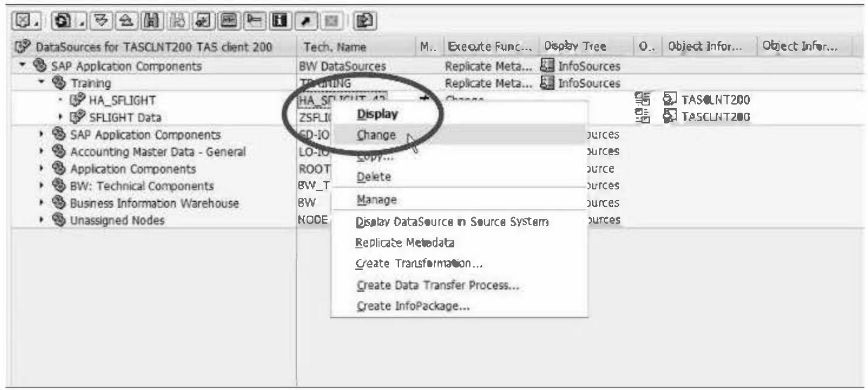


Figure 341: Change

i) Activate (icon) your DataSource HA_SFLIGHT_##. (New SAP HANA Optimized DSO will be created in HANA)

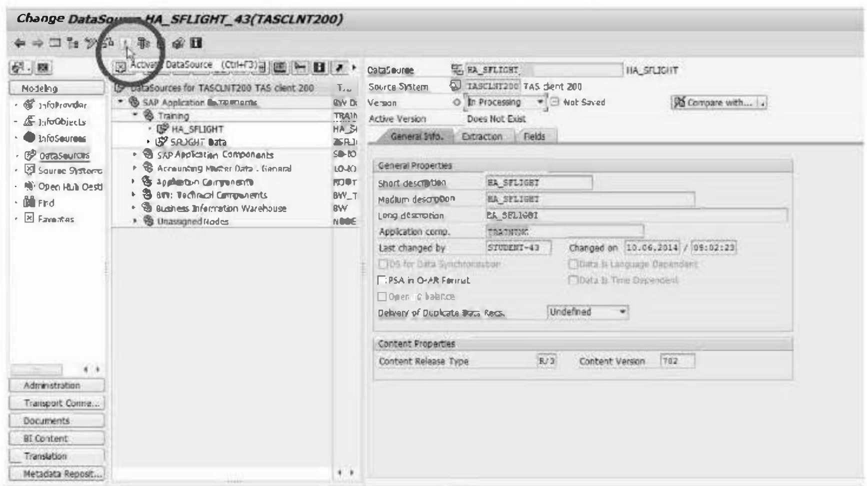


Figure 342: Activate

j) If logon credentials for the SAP HANA system are asked, use the user SYSTEM and the password supplied by the trainer.

Continued on next page

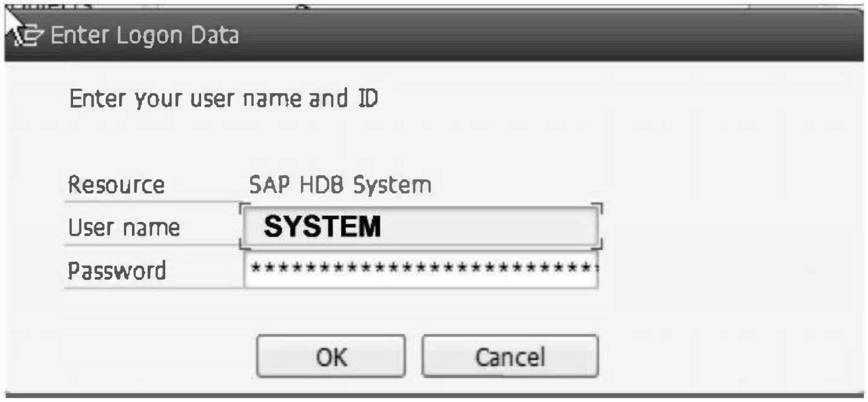


Figure 343: SAP HANA logon details

k) The version should now show as active. (RIght side of screen, just above the row of tabs.)

Continued on next page

Task 4:

Check the new SAP HANA Optimized DSO directly in the SAP HANA Studio.

1. Check the new SAP HANA Optimized DSO and identify the tables in the SAP HANA Studio.

    a) Find and open the schema **DXC<LOCATION>**. in SAP HANA

    Open the tables folder and identify the tables of your new SAP HANA Optimised DSO

    Choose *Open Data Preview* for verifying that the tables are empty.

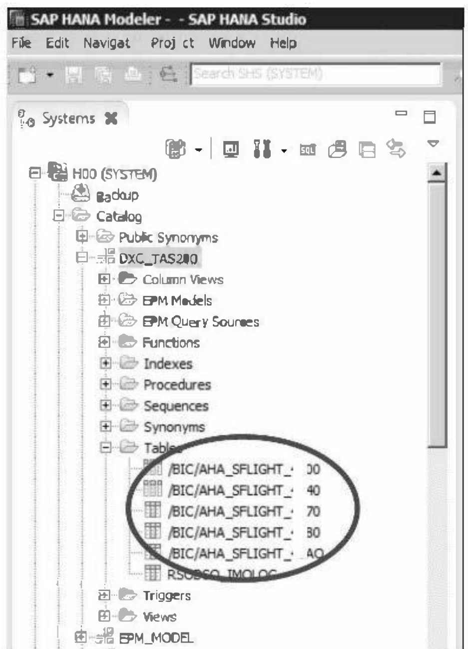


Figure 344: Tables exist in SAP HANA, but still empty

Task 5:

Create an InfoPackage for loading data

1. Create an Infopackage for your datasource.on *Create InfoPackage*. Choose all default values and start the loading process by clicking on *Start* in the last tab.

    Verify the data in BW (load monitor) and on HANA system (table preview on the active table)

Continued on next page

a) Right-click on your DataSource in the SAP system and click on *Create Infopackage....*

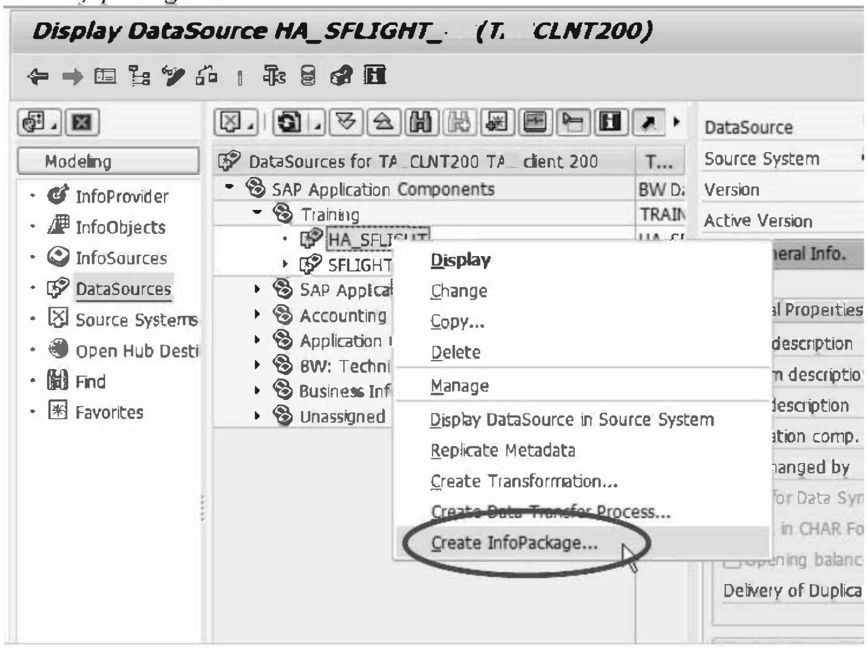


Figure 345: Create Infopackage

b) Choose all default values, and click on ✔

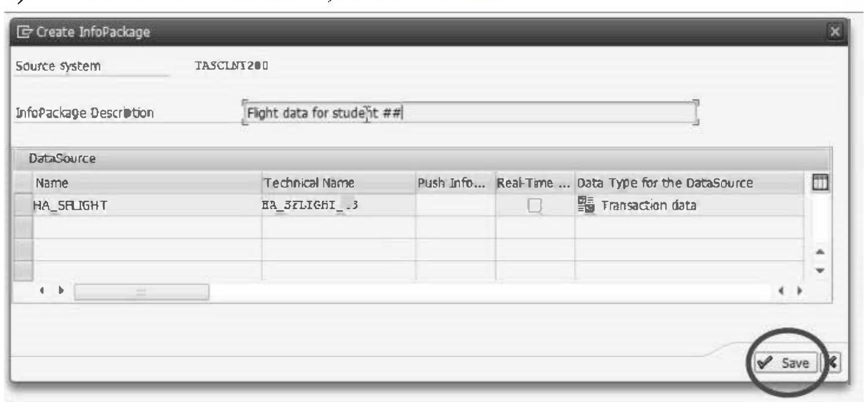


Figure 346: Accept default values

Continued on next page

c) Select the last tab (right side of the screen) called *Schedule*, and start the loading process by clicking on *Start*

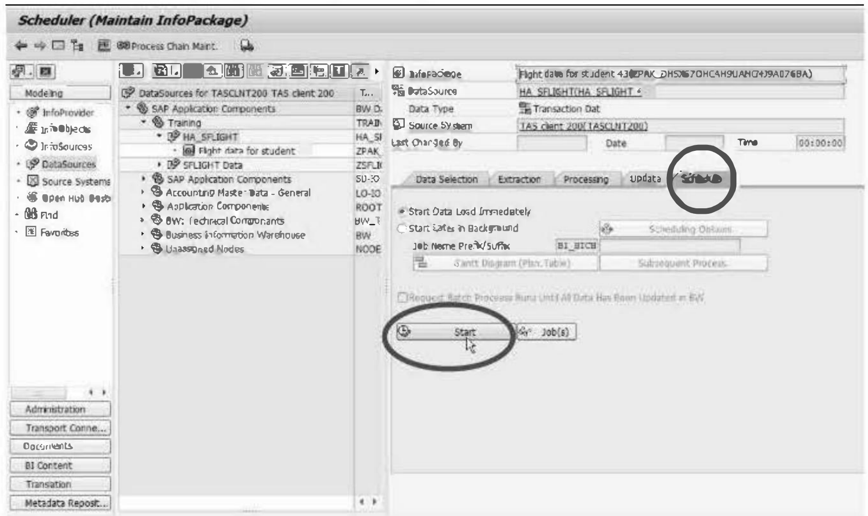


Figure 347: Start the loading process

d) The data is now requested and loaded into SAP HANA

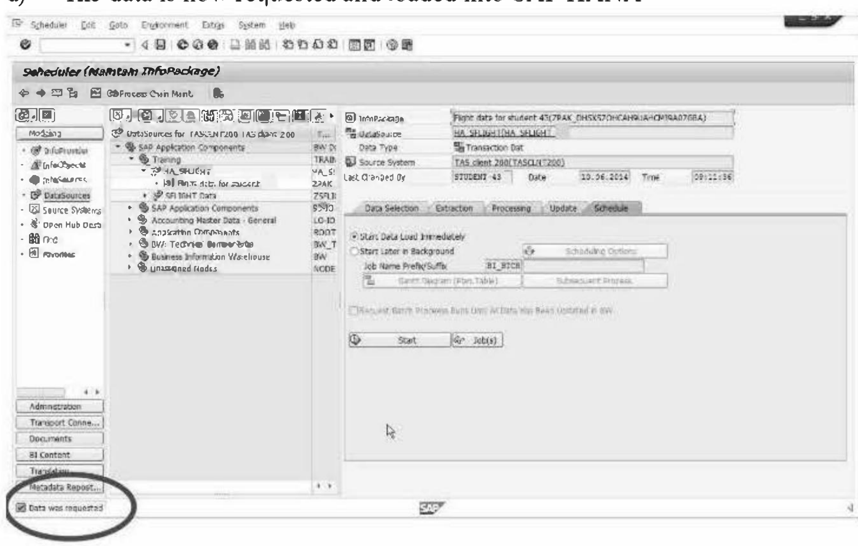
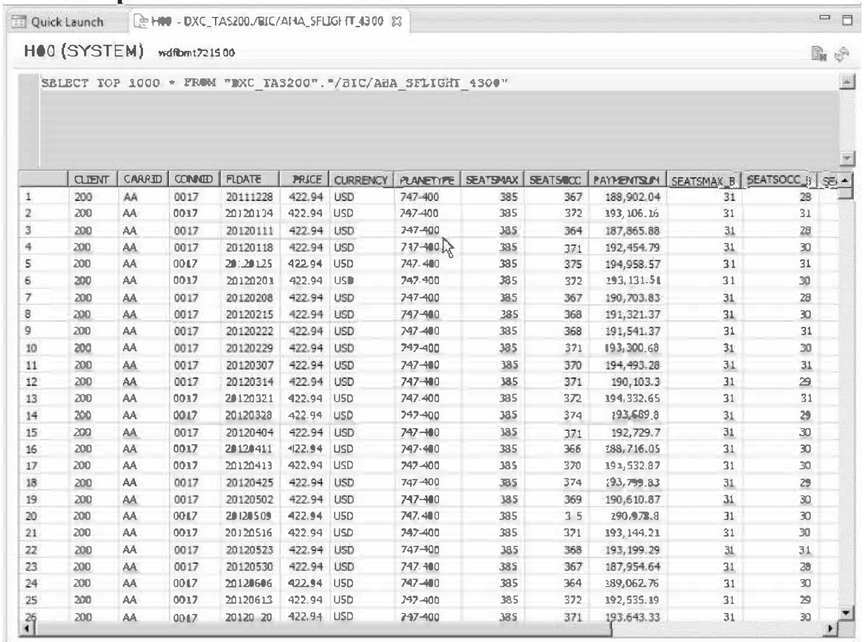


Figure 348: Loading data

Continued on next page

e) You can now verify that the data is loaded into SAP HANA by doing a table preview on the active tables.



	CLIENT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE	SEATSMAX	SEATSOCC	PAYMENTSUM	SEATSMAX_B	SEATSOCC_B	SE
1	200	AA	0017	20111228	422.94	USD	747-400	385	367	188,902.04	31	28	
2	200	AA	0017	20120104	422.94	USD	747-400	385	372	193,106.16	31	31	
3	200	AA	0017	20120111	422.94	USD	747-400	385	364	187,865.88	31	28	
4	200	AA	0017	20120118	422.94	USD	747-400	385	371	192,454.79	31	30	
5	200	AA	0017	20120125	422.94	USD	747-400	385	375	194,958.57	31	31	
6	200	AA	0017	20120201	422.94	USD	747-400	385	372	193,131.51	31	30	
7	200	AA	0017	20120208	422.94	USD	747-400	385	367	190,703.83	31	28	
8	200	AA	0017	20120215	422.94	USD	747-400	385	368	191,321.37	31	30	
9	200	AA	0017	20120222	422.94	USD	747-400	385	368	191,541.37	31	31	
10	200	AA	0017	20120229	422.94	USD	747-400	385	371	193,300.68	31	30	
11	200	AA	0017	20120307	422.94	USD	747-400	385	370	194,493.28	31	31	
12	200	AA	0017	20120314	422.94	USD	747-400	385	371	190,103.3	31	29	
13	200	AA	0017	20120321	422.94	USD	747-400	385	372	194,332.65	31	31	
14	200	AA	0017	20120328	422.94	USD	747-400	385	374	193,689.8	31	29	
15	200	AA	0017	20120404	422.94	USD	747-400	385	371	192,729.7	31	30	
16	200	AA	0017	20120411	422.94	USD	747-400	385	366	188,716.05	31	30	
17	200	AA	0017	20120413	422.94	USD	747-400	385	370	191,532.87	31	30	
18	200	AA	0017	20120425	422.94	USD	747-400	385	374	193,799.83	31	29	
19	200	AA	0017	20120502	422.94	USD	747-400	385	369	190,610.87	31	30	
20	200	AA	0017	20120509	422.94	USD	747-400	385	[illegible]	190,978.8	31	30	
21	200	AA	0017	20120516	422.94	USD	747-400	385	371	193,144.21	31	30	
22	200	AA	0017	20120523	422.94	USD	747-400	385	368	193,199.29	31	31	
23	200	AA	0017	20120530	422.94	USD	747-400	385	367	187,954.64	31	28	
24	200	AA	0017	20120606	422.94	USD	747-400	385	364	189,062.76	31	30	
25	200	AA	0017	20120613	422.94	USD	747-400	385	372	192,535.19	31	29	
26	200	AA	0017	[illegible]	422.94	USD	747-400	385	371	193,643.33	31	30	

Figure 349: Data successfully loaded!

Task 6:

Delete your DataSource in transaction RSA1 and have a look to the coresponding HANA tables. They will be deleted.

1. Right-click on your DataSource in the SAP system and select Delete. Then check it in SAP HANA.

   a) Right-click on your DataSource in the SAP system and click on *Delete*.

Continued on next page

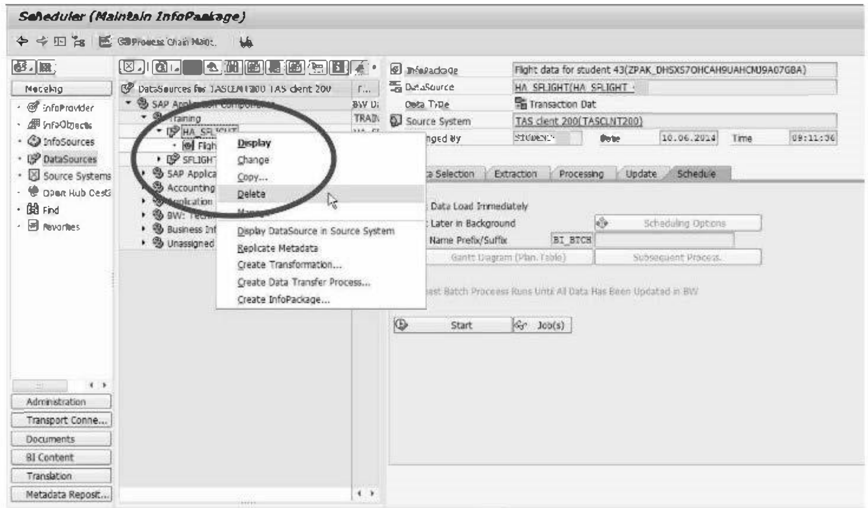


Figure 350: Delete

b) Confirm that you want to delete it

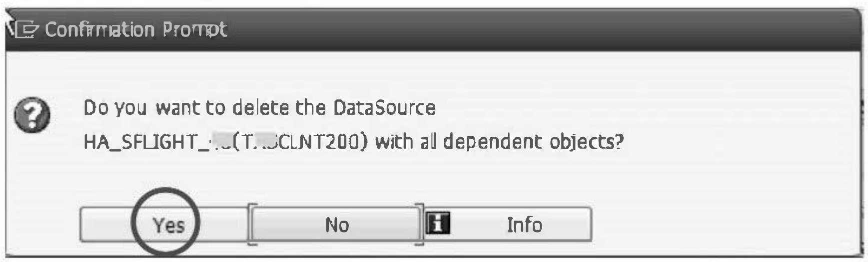


Figure 351: Confirm deletion

c) You might be asked for the SAP logon details again.

Continued on next page

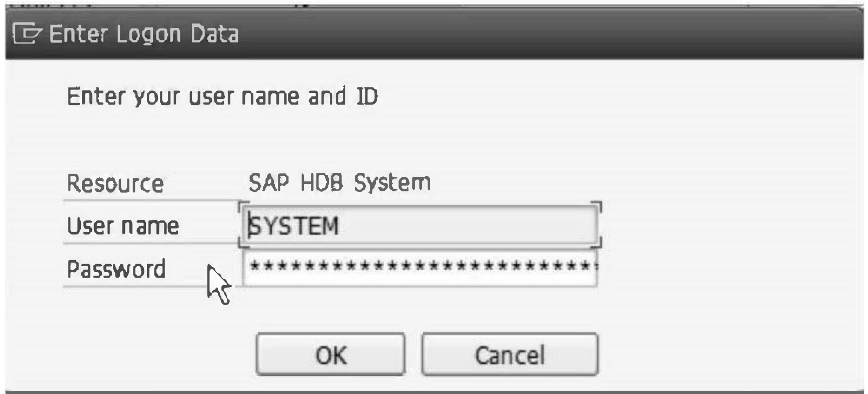


Figure 352: Supply SAP HANA logon details

d) Check that your deleteion request was completed in the SAP system.

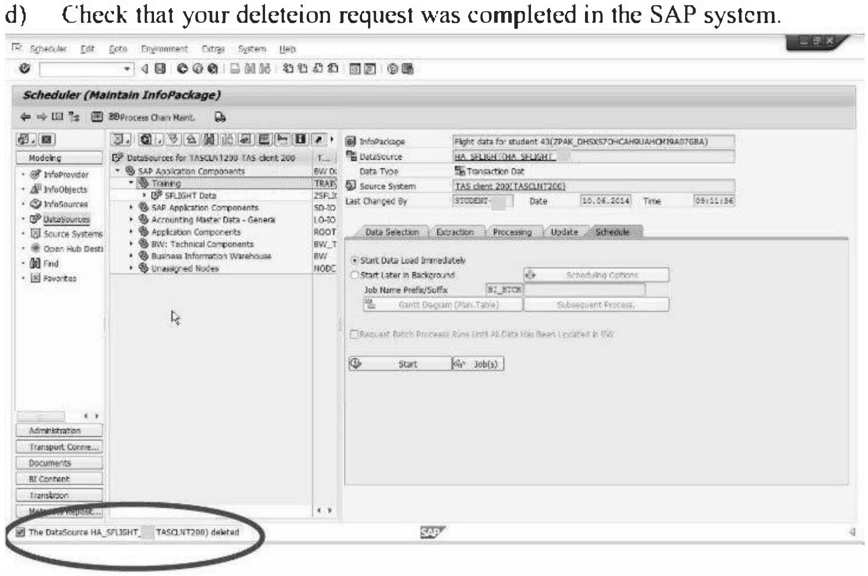


Figure 353: Deletion completed

e) The tables are now also deleted from the SAP HANA system.

Continued on next page

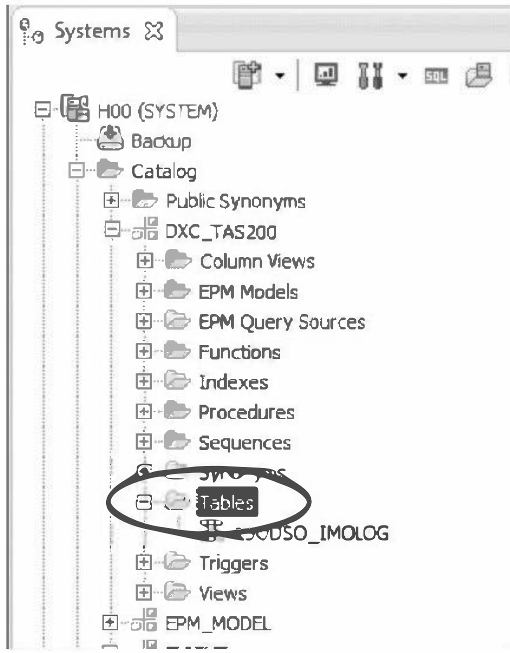


Figure 354: Tables also deleted from SAP HANA

Lesson Summary

You should now be able to:

- Explain SAP HANA Direct Extractor Connection Setup and Configuration

Lesson: DXC Appendix

Lesson Overview

This Appendix covers important SAP Notes for DXC

Lesson Objectives

After completing this lesson, you will be able to:

- know the important DXC Note Numbers

Business Example

You want to implement DXC. Therefore you need to find out about the setup, configuration and pre-requisites

You must read the following SAP Notes before you start the installation. These SAP Notes contain the most recent information on the installation, as well as corrections to the installation documentation.

All relevant information related to setup & configuration is provided by

Note 1665602 - Setup & Config: SAP HANA Direct Extractor Connection (DXC)

Note 1583403 - Direct extractor connection to SAP HANA

Figure 355: Setup and Configuration - Relevant notes

Make sure that you have the up-to-date version of each SAP Note, which you can find on SAP Service Marketplace at http://service.sap.com/notes.

SAP Note Number	Title	Description
1583403	Direct extractor connection to SAP HANA	Main note for setup steps required in the source SAP Business Suite system.
1670518	SAP HANA Direct Extractor Connection:	Monitoring Provides information how to monitor SAP HANA Direct Extractor Connection (DXC), in particular the Activation processing for SAP HANA Optimized DataStore Objects
1688750	DataSource: Reading a property in the source system	Apply this note to the source SAP Business Suite system only if you have the "sidecar" scenario described in section Appendix – DXC System Landscape Variants: The "Sidecar" Approach.
1701750	DataSource Secondary Index on the PSA	If your DataSource is missing a key, apply this note to any BW systems connected to the SAP Business Suite system you are using with DXC
1677278	DataSource: Changing the Key Definition (A version)	Provides a report where you can define a semantic key for any DataSources that are missing keys. DataSources without keys will cause an error when you try to generate the SAP HANA Optimized DSO. Before applying this not to your SAP Business Suite system, first apply SAP note 1701750 to any BW systems connected to

SAP Note Number	Title	Description
		the SAP Business Suite system you are using with DXC.
1710236	SAP HANA DXC: DataSource Restrictions	Lists specific DataSources not supported by DXC.
1714852	Troubleshooting SAP HANA DXC issues	Guidance for troubleshooting DXC issues

Lesson Summary

You should now be able to:

- know the important DXC Note Numbers

Lesson: Appendix: DXC Sidecar Variation

Lesson Overview

- DXC Sidecar Variation

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the DXC Sidecar Variation

Business Example

You have an older Business Suite system thereofre you cannot use the embedded BW Data Sources for DXC. In that case you want to use the side car variation.

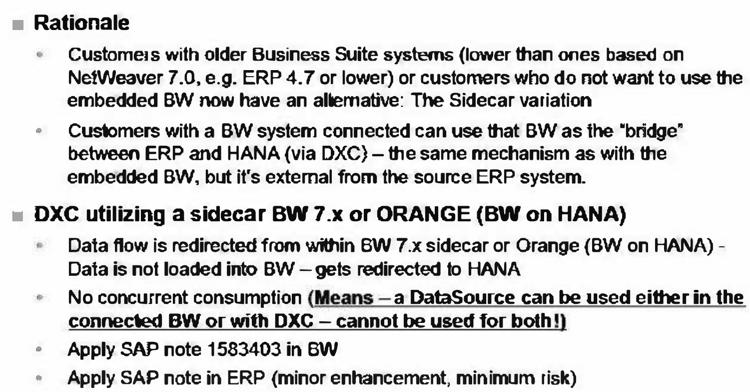


Figure 356: Appendix: DXC Sidecar Variation SAP HANA Direct Extractor Connection: Sidecar

The default configuration relies on the "embedded BW" which exists inside of SAP NetWeaver 7.0 or higher (e.g. ECC 6.0). However, in some cases, customers may be interested in implementing DXC with an SAP Business Suite system that is older, and therefore not based on SAP NetWeaver 7.0 or higher (e.g. 4.6C). Another use case might be that the "embedded BW" is already in use. As a consequence, the customer might be reluctant to use it for this purpose; also there may be some general preference to avoid the use of the embedded BW system on an SAP Business Suite system, even though it is primarily used for scheduling and monitoring of extraction jobs in that scenario.

In order to enable DXC when such conditions exist, DXC can be implemented with a "sidecar" approach. This means, that instead of using the "embedded BW" inside the SAP Business Suite system, a separate connected BW system can be used as an intermediary system for scheduling and managing the extraction job in the connected SAP Business Suite system, which sends the extracted data directly to SAP HANA. Extracted data will not be loaded into the connected SAP BW system; instead the data flow will be redirected to the SAP HANA system.

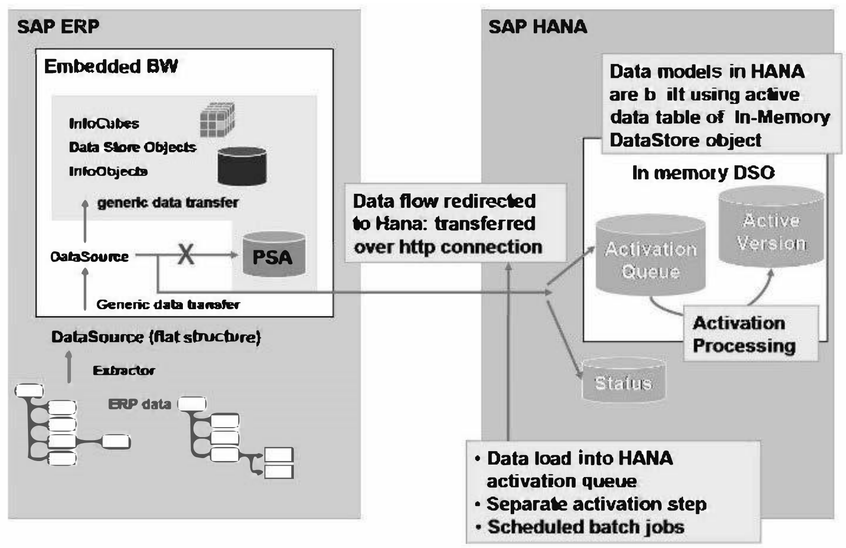


Figure 357: Appendix: DXC Sidecar Variation SAP HANA DXC Concept: Illustration Embedded BW

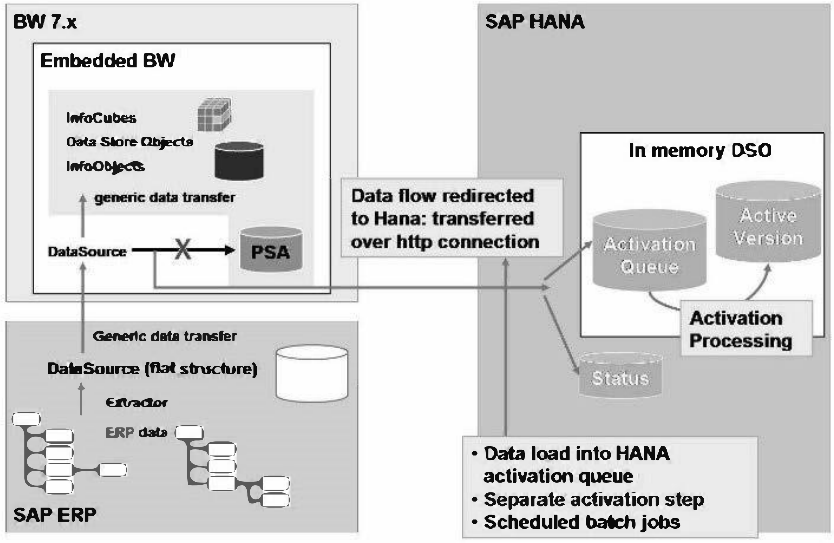


Figure 358: Appendix: DXC Sidecar Variation SAP HANA DXC Concept: Illustration Standalone BW

Another similar variation is available to customers running "SAP NetWeaver BW Powered by SAP HANA" (aka BW on HANA).

This illustration depicts another variation of the "sidecar" approach.

Lesson Summary

You should now be able to:

- Explain the DXC Sidecar Variation

Unit Summary

You should now be able to:

- Explain an additional data acquisition technique for working with data from SAP Business Suite systems that has been added to the existing techniques for HANA data acquisition.
- Explain the SAP Business Content DataSource Extractors
- Explain the SAP HANA Direct Extractor Connection in detail
- Explain the Comparison with other SAP HANA Data Acquisition Techniques
- Explain SAP HANA Direct Extractor Connection Setup and Configuration
- know the important DXC Note Numbers
- Explain the DXC Sidecar Variation

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Unit 8

Uploading Data from a Flat File

Unit Overview

This Unit shows you the possibility of uploading Flat - File into SAP HANA

Unit Objectives

After completing this unit, you will be able to:

- Understand when to use Flat File data load functionality
- How to load data from Flat Files into the SAP HANA Database

Unit Contents

Lesson: Loading Flat File Data into SAP HANA

Lesson Overview

One of the options available for data provisioning is to simply upload data from flatfiles. The lesson shows you the steps required to upload your own data from a csv file.

Lesson Objectives

After completing this lesson, you will be able to:

- Understand when to use Flat File data load functionality
- How to load data from Flat Files into the SAP HANA Database

Business Example

You want to upload Sales Organisation texts from a flat file to an SAP HANA table, so you can use the data in modeling afterwards.

Uploading data from flat files

It is possible to upload data from flat files, available at client file system, to SAP HANA database

If the required table for loading the data does not exist in SAP HANA database, it's necessary to create a table structure that is based on the flat files

The application suggests the column names and data types for the new tables and it's possible to edit them

The new table always has a 1:1 mapping between the file and table columns

When loading new data in the table, it gets appended to the existing data

The application does not allow to overwrite any column or change the data type of existing data

The supported file types are: .csv, .xls, and .xlsx

Especially suited for Proof of Concepts or projects where only an one-time data load is required

+ Quick and easy data load

– No delta logic available; no transformation capabilities → 1:1 mapping only

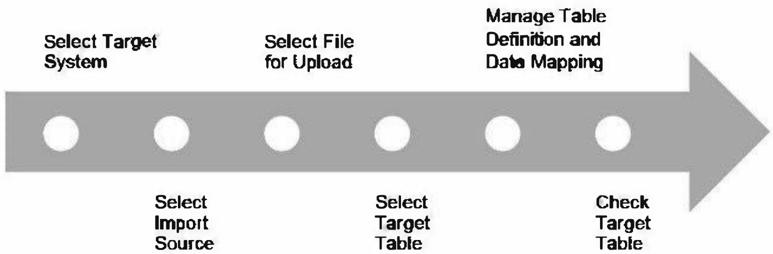


Figure 359: Process Flow: Uploading Data from Flat Files

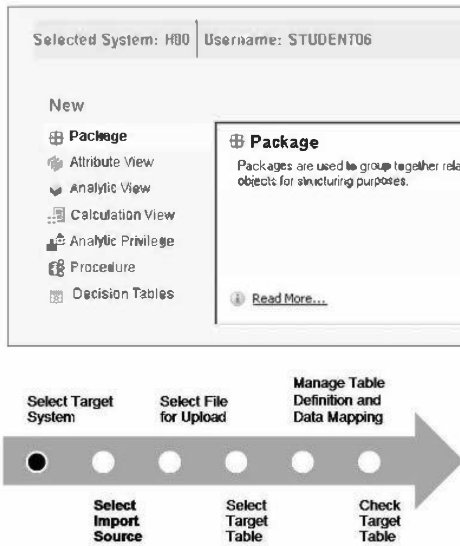


Figure 360: Select Target System

In the *Quick Launch* screen, select the Target System where the data should be imported.

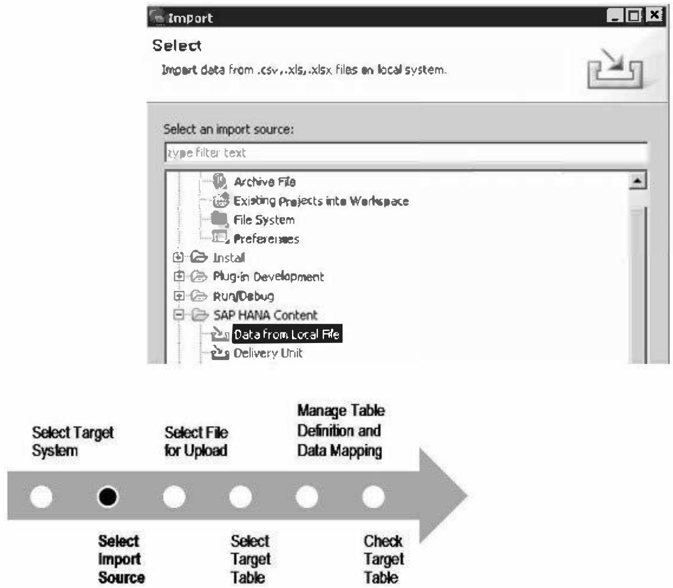


Figure 361: Select Import Source

- In the file menu, choose *Import*
- Expand the SAP HANA Content directory
- Select *Data From Local File* and choose Next

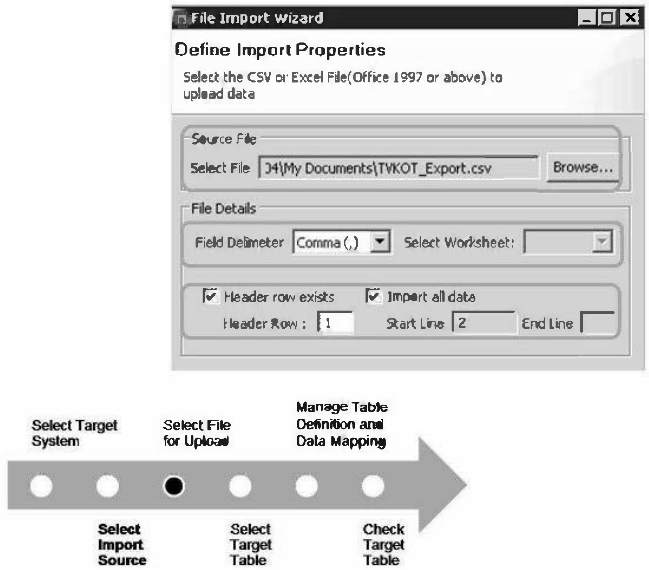


Figure 362: Select File for Upload

- In the Flat File Upload screen, browse for the file which should be uploaded into SAP HANA database
- If a .xls or .xlsx file has been selected, choose the corresponding worksheet
- If a .csv File has been selected, select a delimiter
- If a header row exists in the flat file, select Header row exists and enter row number
- If only a specific row range should be relevant for the import, remove check for Import all data and enter the start / end line

Note: A delimiter is used to determine columns and pick correct data against them. In a .csv file, the accepted delimiters are: , ;

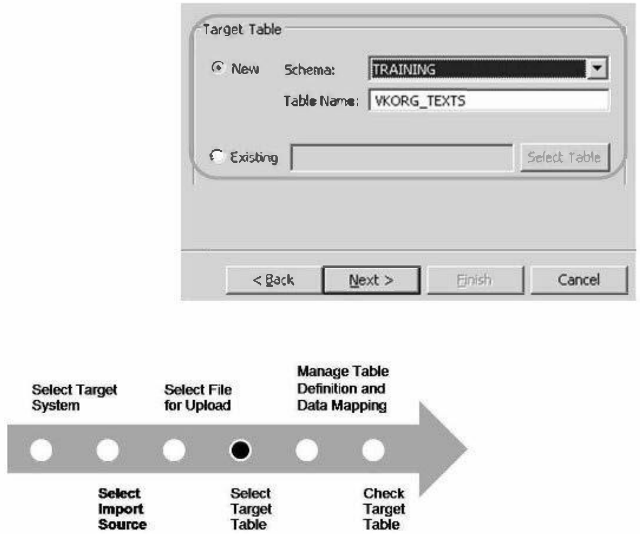


Figure 363: Select Target Table

- For the Target Table, two options are available:
  - New: When selecting New, a new table with the name entered will be generated within the schema chosen.
  - Existing: When selecting Existing, data will be appended to an existing table.
- Choose Next

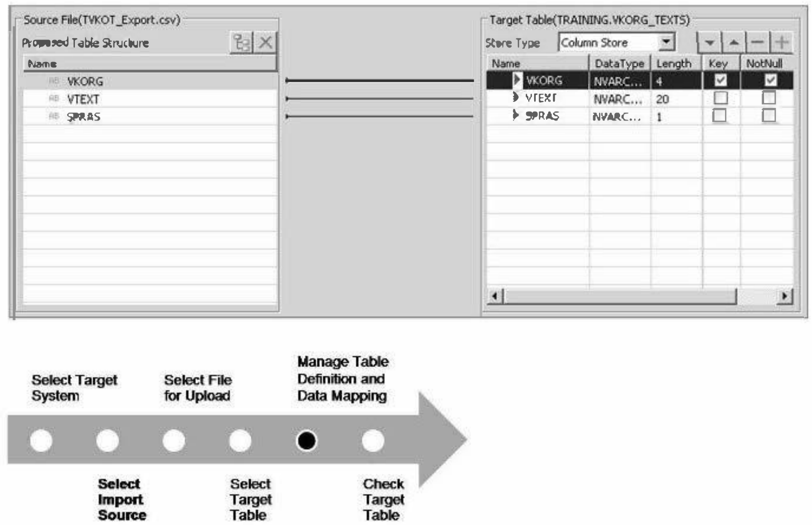


Figure 364: Manage Table Definition and Data Mapping

- In the Manage Table Definition and Data Mapping screen it's possible to map the source and the target columns
- The application proposes a mapping structure automatically based on the naming
- Additionally it's required to select a Key

Note: Only 1:1 column mapping is supported. Additionally, it's possible to edit the table definition by changing the store type, data types, renaming, adding or deleting columns.

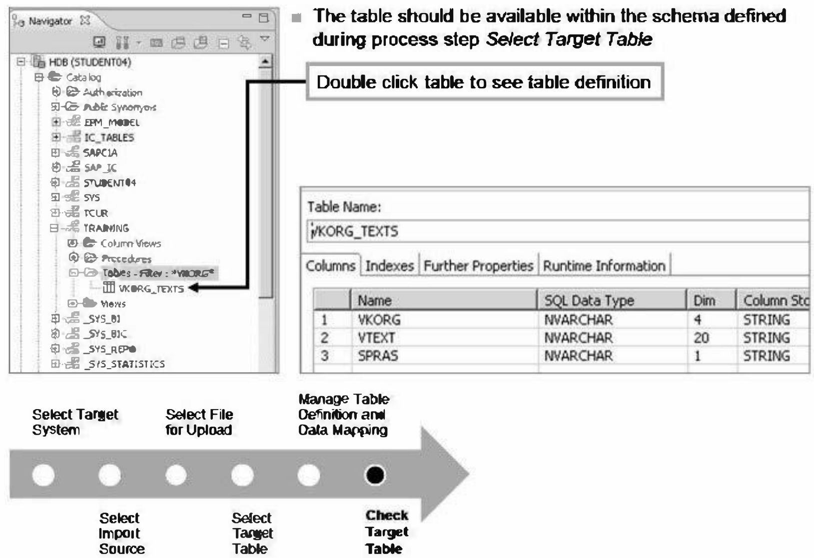


Figure 365: Check Target Table

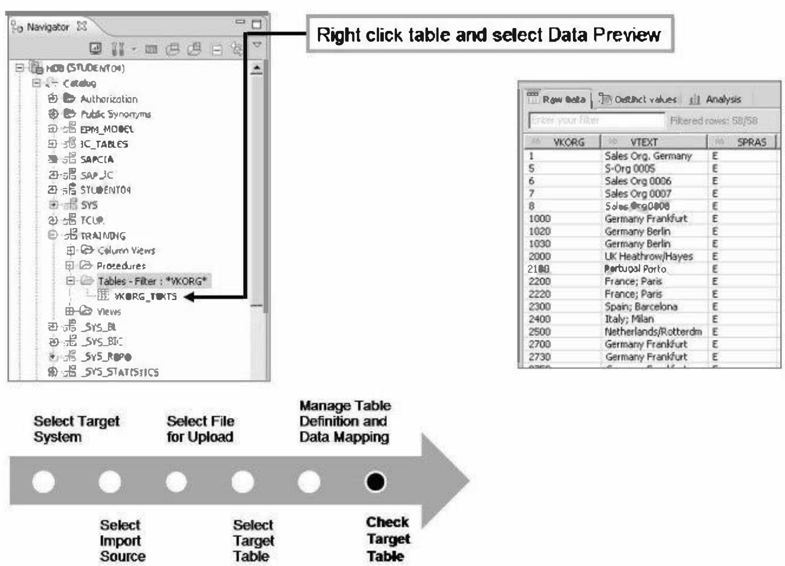


Figure 366: Check Target Table

Exercise 6: Uploading a flatfile into the HANA database

Exercise Objectives

After completing this exercise, you will be able to:

- Learn how to upload a structured flat file into a HANA database table.

Business Example

You got a CSV file with the description of the sales organisations and you want to upload the data into a new database table.

Task 1:

In this exercise you will upload a flat file with a list of descriptions of sales organisations into the HANA database.

1. Find the VKORG_TEXTS.CSV file that was uploaded to the network
2. Import the CSV file and create a new table called VKORG_TEXTS in the schema STUDENT## (where ## is your student number) with this data file

Result

You have created a new database table VKORG_TEXTS in the schema STUDENT## (where ## is your student number) and you populated it with the data from the flat file.

Task 2:

Now you will check the data in the newly created table.

1. Open your schema STUDENT## and refresh the display. Then check the table definition and table content of the newly created table.

Result

You have successfully imported a flat file into SAP HANA.

Solution 6: Uploading a flatfile into the HANA database

Task 1:

In this exercise you will upload a flat file with a list of descriptions of sales organisations into the HANA database.

1. Find the VKORG_TEXTS.CSV file that was uploaded to the network

   a) Click on the Windows Start button. Then click on Program Files. On the menu, you will find an item marked “Student”. Select this menu option.

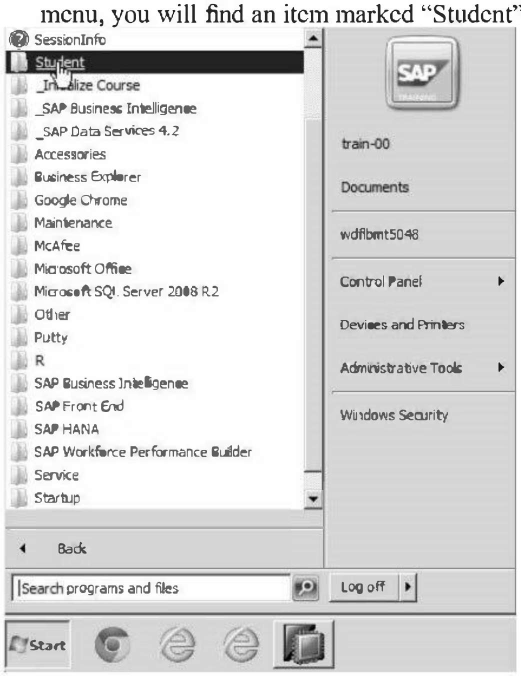


Figure 367: Student Folder shortcut

   b) You will be asked for a username and password. Enter **hanastudent** as the username, and **hanareadonly** as the password.

Continued on next page

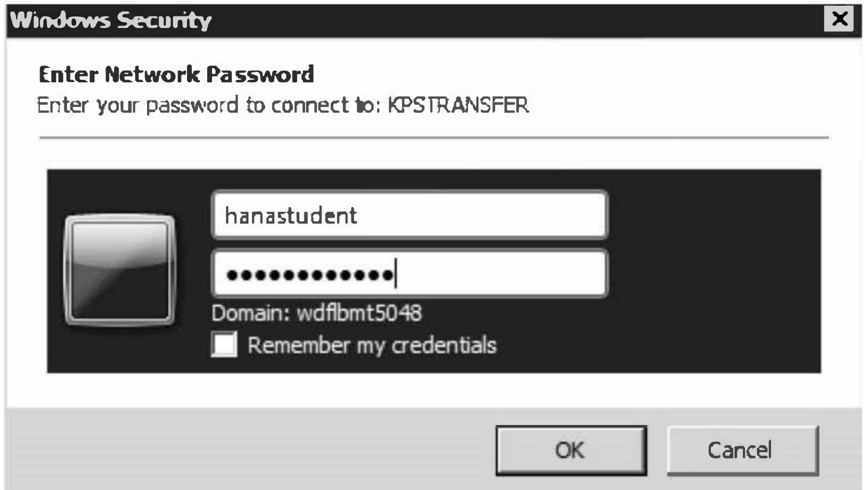
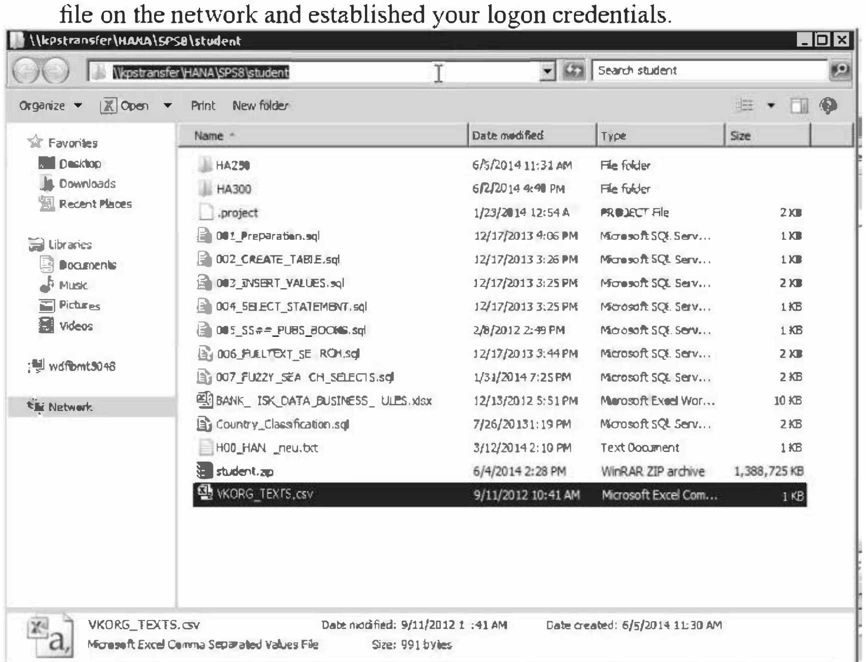


Figure 368: Username and password

c) The CSV file will be listed in the student folder. You have now found the file on the network and established your logon credentials.



Continued on next page

Figure 369: Find the CSV file

2. Import the CSV file and create a new table called VKORG_TEXTS in the schema STUDENT## (where ## is your student number) with this data file

   a) Open the Quick Launch page in the SAP HANA Studio, and click on the Import option (on the right side). If you do not see the Quick Launch page, make sure you are in the Modeler perspective. Alternatively you can click on the File menu, and then select the Import menu option.

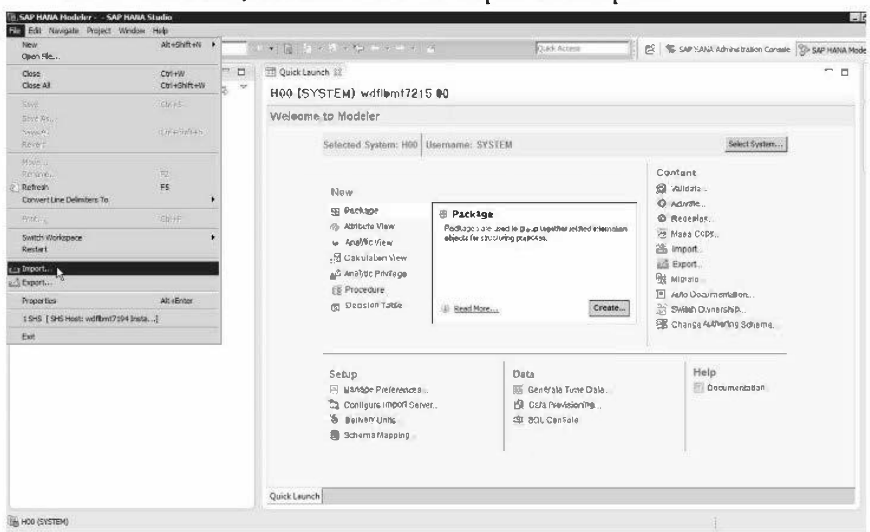


Figure 370: Import

   b) Choose SAP HANA Content –> Data from Local File as an import source. Then click on Next

Continued on next page

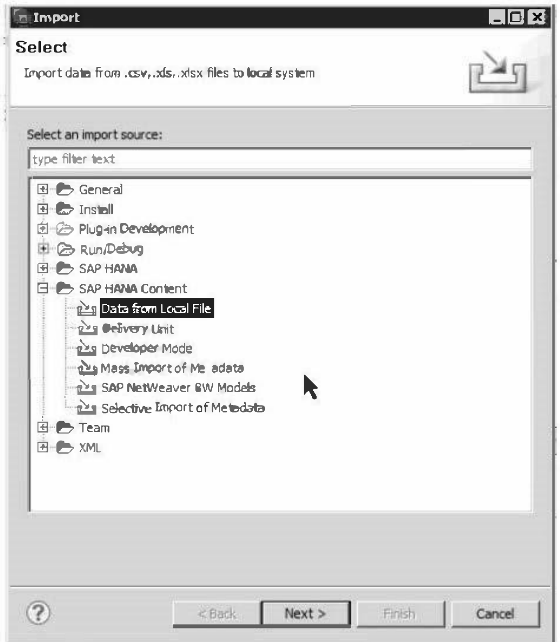


Figure 371: Data from Local File

c) Select the target system where you want to load the data. In this case it will be system H00. Then click on Next

Continued on next page

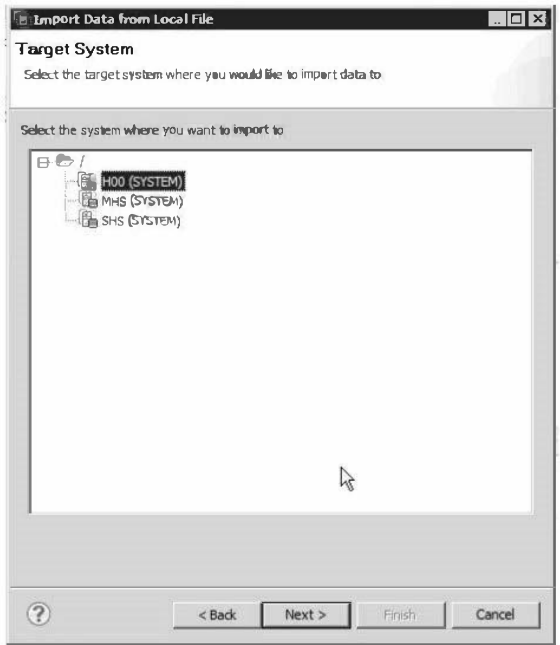


Figure 372: Target system is H00

d) Select the source file by clicking on the Browse button. Under Documents you will find the CSV file in the '*HANA Student*' folder. Select the VKORG_TEXTS.csv file, and click on the Open button.

Continued on next page

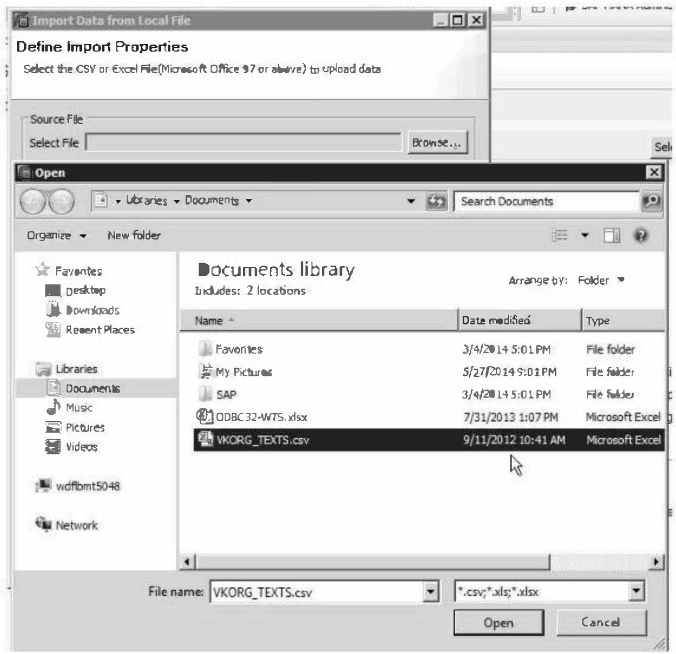


Figure 373: Find the CSV file

c) Now set the import properties for this file.

- Set the field delimiter as a comma
- Tick the "Header row exists" box
- Ensure the Header Row start with row number 1, and the Start Line at 2
- You will create a new table in your STUDENT## schema (where ## is your student number)
- Name your new table VKORG_TEXTS_## (where ## is your student number)

Then click on Next

Continued on next page

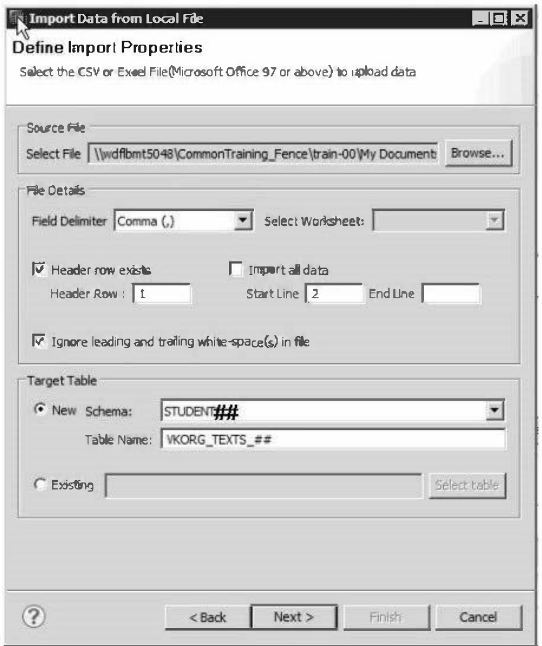


Figure 374: Specify the import file properties

f) On the next screen you will manage your data mappings. In this case you will keep it as it is. You do however have to specify the field VKORG as primary key of the table.

Mark the check box in the key column of the row WKORG.

Click on Next

Continued on next page

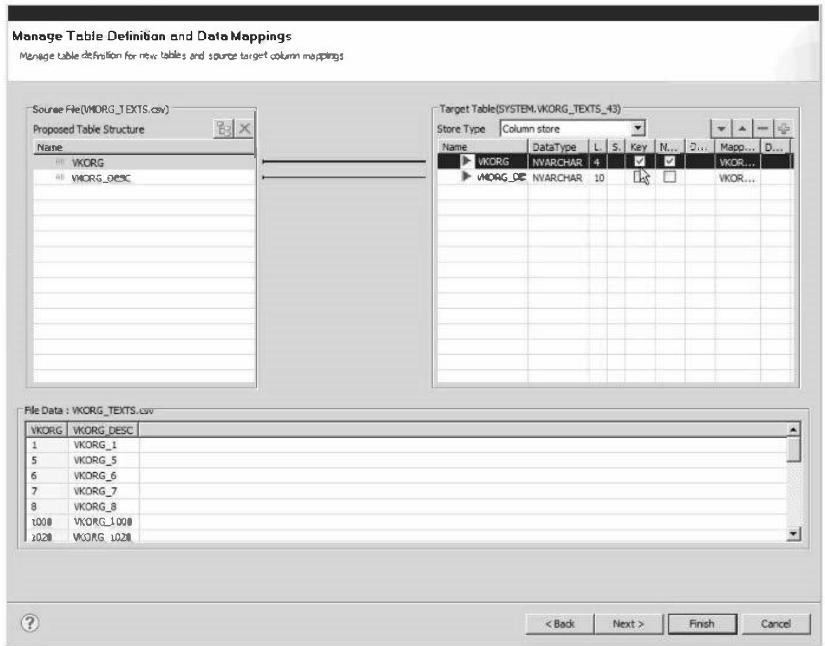


Figure 375: Manage the data mappings

g) On this last screen you can confirm the import. Click on Finish to create the new table and import the data.

Continued on next page

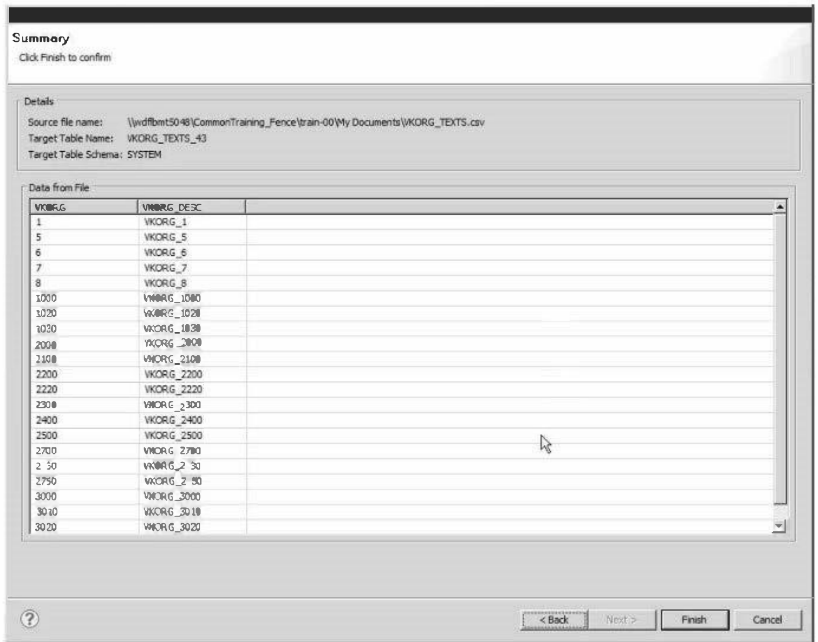


Figure 376: Confirm the import

h) You have now successfully created a new table and imported the data from a flat file into that table. In the next task we will verify the data import.

Result

You have created a new database table VKORG_TEXTS in the schema STUDENT## (where ## is your student number) and you populated it with the data from the flat file.

Task 2:

Now you will check the data in the newly created table.

1. Open your schema STUDENT## and refresh the display. Then check the table definition and table content of the newly created table.

   a) Open theH00 system in the HANA Studio systems view tab.

      Open the Catalog folder. Then look for your STUDENT## schema. Where ## is your student number.)

      Open your STUDENT ## schema, and then open the tables area.

Continued on next page

Right-click on the tables folder and click on Refresh (or press F5)

You should now see your VKORG_TEXTS_## table (where ## is your student number)

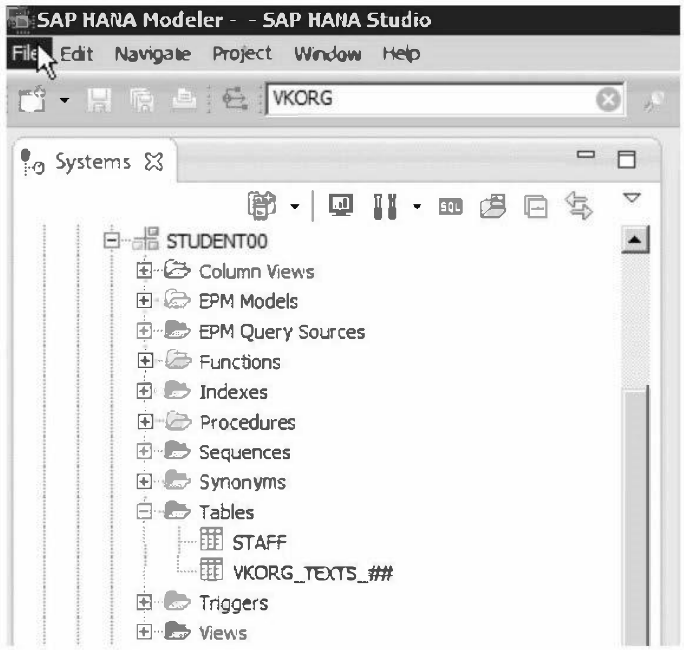


Figure 377: Find the new table in your STUDENT## schema

Double click on the VKORG_TEXTS table. This will open the table definition, and display the properties of the table.

b)

Continued on next page

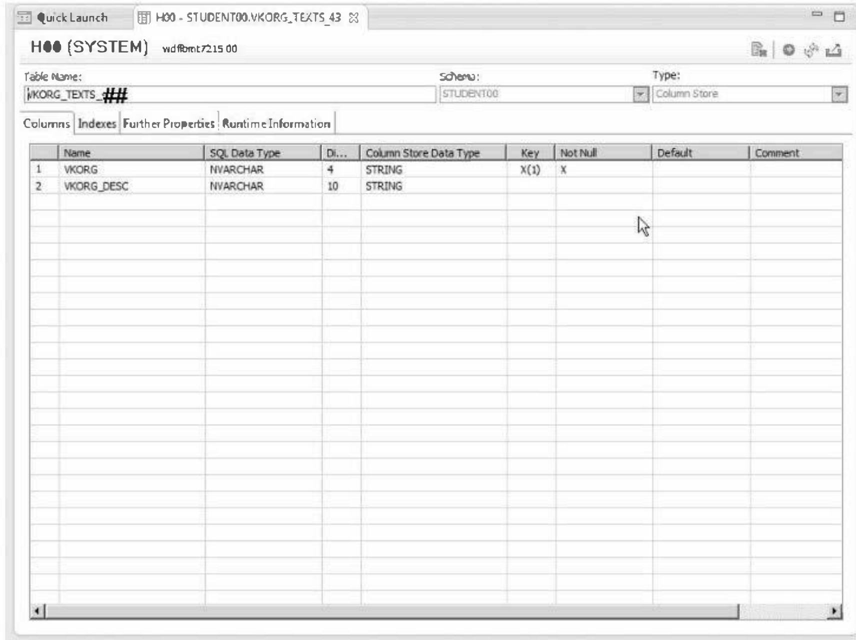


Figure 378: The table definition

c) Right-click on the table name, and select Open Content

Continued on next page

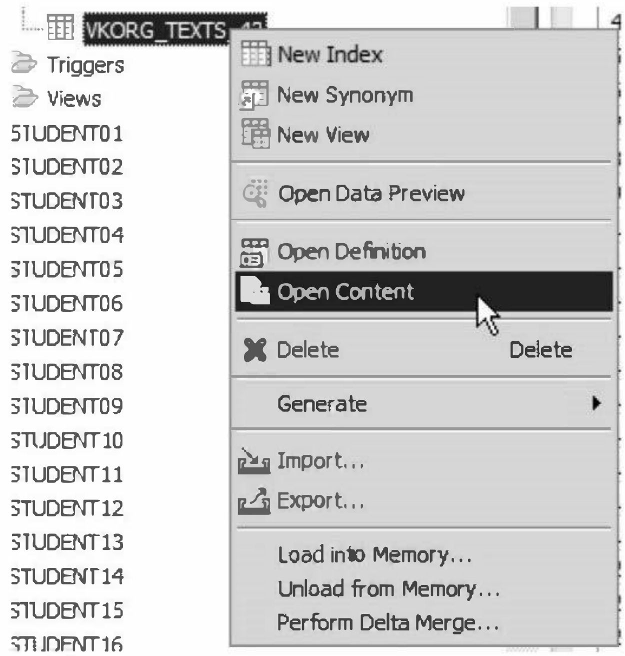


Figure 379: Open the table content

d) You can now see the content of the table. It is identical to the content of the flatfile.

Continued on next page

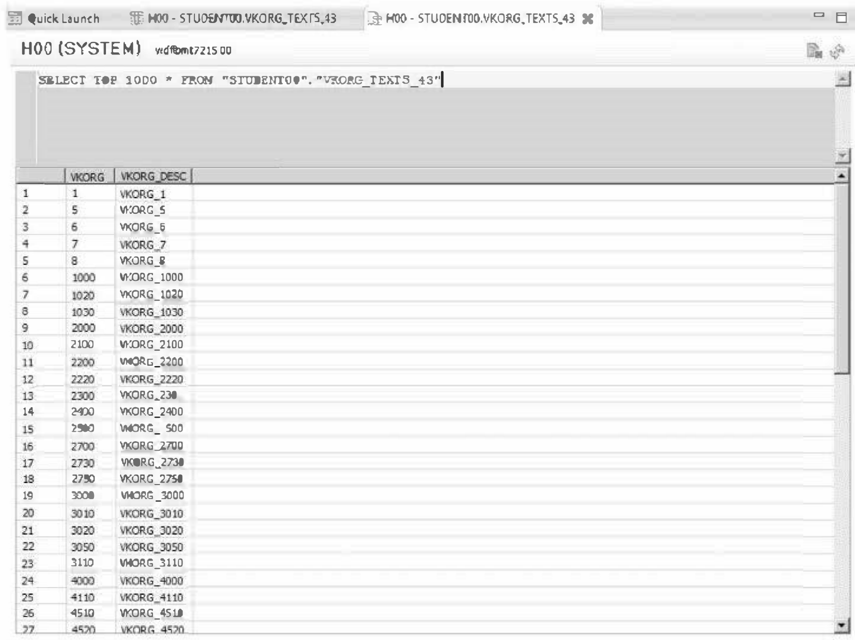


Figure 380: Success!

Result

You have successfully imported a flat file into SAP HANA.

Lesson Summary

You should now be able to:

- Understand when to use Flat File data load functionality
- How to load data from Flat Files into the SAP HANA Database

Unit Summary

You should now be able to:

- Understand when to use Flat File data load functionality
- How to load data from Flat Files into the SAP HANA Database

Unit 9

Optional Unit; SAP Replication Server (SRS)

Unit Overview

This Unit gives you an overview of SAP Replication Server

Unit Objectives

After completing this unit, you will be able to:

- Explain the key benefits of SAP Replication Server
- List the supported primary and replicate RDBMS
- Describe the architecture of SAP Replication Server
- Understand how to connect SAP Replication Server with an SAP HANA system

Unit Contents

Lesson: SAP Replication Server

Lesson Overview

This lesson gives an overview of SAP Replication Server, a log-based database replication tool, and explain how it can be used to provision data into an SAP HANA database.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the key benefits of SAP Replication Server
- List the supported primary and replicate RDBMS
- Describe the architecture of SAP Replication Server
- Understand how to connect SAP Replication Server with an SAP HANA system

Business Example

In the context of an SAP HANA implementation, you have decided to implement SAP Replication Server to serve as your data provisioning tool. You want to know more about this technology, and also discuss its future development within the SAP product offering.

SAP Sybase Replication Server Introduction

Replication Server is a sophisticated transactional data movement product that moves and synchronizes data across the enterprise, without geographical distance limitation, to meet demanding requirements in the enterprise such as guaranteed data delivery, real-time business intelligence and zero operational downtime.

Replication Server facilitates this by non-intrusively handling data at the source and target, including Sybase, Oracle, Microsoft, and IBM, while ensuring high performance and transactional integrity.

- Log-based replication process
  - Non-intrusive
  - Very high performance
- Improve recovery, resumption times and minimize downtime
  - Bi-directional replication
  - Standby DB is always available and can be used for read-only report server
- Fresh data to enable timelier decision
  - Run ressource-instensive reports on reporting servers without impacting OLTP systems
  - Reduce information latency for reporting and optimize batch reporting
- Real time data sharing and synchronization
  - Facilitates decentralized business operations
  - Enables remote applications to access data locally for improved pefonnance

Figure 381: SAP Replication Server – Main Objectives

One key feature of SAP Replication Server is that it relies on a log-based replication technique: the Changed Data Capture (CDC) is not done against the data volumes of the source database, but instead by reading directly the database log.

This log-based approach reduces the workload that the replication process brings to the source database, thus enhancing the availability of this system.

- Data distribution and migration
  - Distribute: move centralized data to operational applications
  - Share: share data between operational applications
  - Synchronize: maintain consistency in overlapping data values
  - Migrate: move from older version of database platform to newer
- Real-time decision support
  - Create Operational Data Stores (copy of OLTP production systems for daily reporting)
  - Real-time loading of data warehouses (SAP HANA, SAP IQ, SAP ASE, Oracle, Microsoft, IBM)
- Real time data sharing and synchronization
  - Enable business continuity in case of site-wide disaster
  - Maintain application availability during planned/unplanned downtime

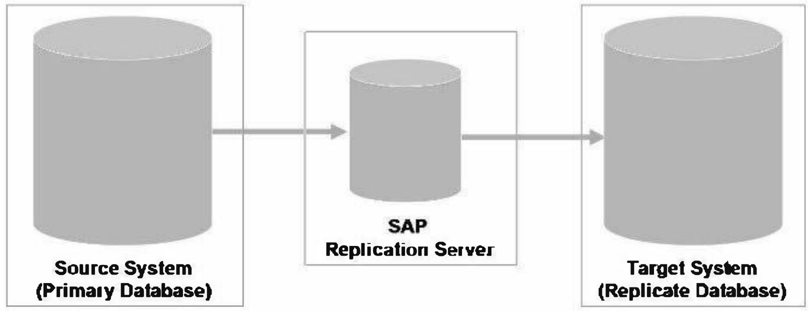
Figure 382: SAP Replication Server – Example of Scenarios

Note: SAP Replication Server supports a lot of scenarios, such as

- Bi-directional replication
- Support for Disaster Recovery
- One-to-Many / Many-to-One replication

However, in scenarios including SAP HANA, only the **replication database** can run on SAP HANA, not the source database.

Supported Databases and Server Architecture



- SAP ASE
- Oracle
- MS SQL server
- IBM DB2/UDB
- IBM DB2/OS390

- SAP HANA
- SAP ASE
- SAP IQ
- SAP ESP
- Oracle
- MS SQL server
- IBM DB2/UDB
- IBM DB2/OS390

Figure 383: SAP Replication Server – Supported Databases

In the SAP Replication Server terminology, the source system is called *Primary Database*, and the target system *Replicate Database*.

The above figure lists the types of databases that are supported by SAP Replication Server, either as the Primary or the Replicate Database.

You will notice the wide range of supported source database, meaning that SAP Replication Server is able to read and interpret the log's content of many database vendors.

Note: Depending on the Primary and Replicate database, some RDBMS features (data types, and so on) may not be supported. Additionally, not all the configurations allow the replication of Data Definition Language (DDL) statements.

We recommend that you consult the SAP Replication Server documentation available on the SAP Help Portal, at http://help.sap.com/replication-server.

SAP Replication Server with SAP HANA

- SAP Replication Server minimum version for SAP HANA support: 15.7 SP100
- Heterogeneous materialization (Initial load)
  - Pre-requisite before you can replicate transactions
  - Optimized for numerous tables and big data volumes (massive parallelism)
- Leverages native HANA ODBC driver
  - Strong bulk insert/delete capabilities
- Provides Data Assurance utility for ASE and Oracle DB as primary database

Figure 384: SAP Replication Server – SAP HANA as the Replicate Database

SAP HANA is only supported as the replicate database, and from SAP Replication Server version 15.7 SP100 onwards.

Additionally, the replication of Data Definition Language (DDL) statements is not currently supported.

Summary of SAP HANA Provisioning methods

	Flat File Upload	ETL-based (E.g. SAP BO Data Services)	SAP Landscape Transformation (SLT)	SAP Sybase Replication Server	Direct Extractor Connection (DXC)
Use Case	Prototyping and Testing	Productive Usage	Productive Usage	Productive Usage	Productive Usage
Focus	Agility	Extraction and Transformation	Real-time	Real-time	Reusability of SAP Content
Number of souces	csv, xls	high	high	limited	Limited to SAP Business Suite
Transformation capatilities	None (1:1)	Full range: From simple functions to very complex transformation logic	Default 1:1, conditional filtering and little transformations possible	None (1:1)	Limited to logic defined within function module of SAP DataSource

Figure 385: Summary of SAP HANA Data Provisioning

Lesson Summary

You should now be able to:

- Explain the key benefits of SAP Replication Server
- List the supported primary and replicate RDBMS
- Describe the architecture of SAP Replication Server
- Understand how to connect SAP Replication Server with an SAP HANA system

Unit Summary

You should now be able to:

- Explain the key benefits of SAP Replication Server
- List the supported primary and replicate RDBMS
- Describe the architecture of SAP Replication Server
- Understand how to connect SAP Replication Server with an SAP HANA system

SAP Certification Books ,Server access & Training Email:info@araniconsulting.com or visit www.araniconsulting.com

Course Summary

You should now be able to:

- Use SAP System Landscape Transformation **(SLT)**
- Use SAP Data Services **(DS)**
- Use Smart Data Access **(SDA)** ,
- Use SAP HANA Direct Extractor Connection **(DXC)**
- Load flat files as one way of provisioning data for your SAP HANA system.
- Understand SAP Replication Server **(SRS)**
- Smart Data Integration **(SDI)** and Smart Data Quality **(SDQ)**as part of Enterprise Information Management **(EIM)**
- Smart Data Streaming (SDS)

Appendix 1

Abbreviations

Apache Spark	Apache Spark is an open-source cluster computing framework .In contrast to Hadoop's two-stage disk-based MapReduce paradigm, Spark's in-memory primitives provide performance up to 100 times faster for certain applications
BW	Business Warehouse
CRM	Customer Relationship Management
CTS+	Enhanced Change and Transport System. CTS+ allows using transport mechanisms from ABAP (CTS) for Java objects as well.
DBA Cockpit	The DBA Cockpit is a platform-independent tool that you can use to monitor and administer your database.
DDL	Data Definition Language
DMIS	Data Migration Server
ECC	Enterprise Central Component
ERP	Enterprise Resource Planning
EWA	Early Watch Alert

Hadoop	Hadoop is a set of algorithms (an open-source software framework written in Java) for distributed storage and distributed processing of very large data sets (Big Data) on computer clusters built from commodity hardware
IMDB	In-Memory Database
JAVA	Programming Language
JDBC	Java Database Connectivity – Java Standard Edition platform
NLS	Near Line storage
ODBC	Open Database Connectivity – is a standard
OLAP	Online Analytical Processing
PAK	Planning Application KID
RDBMS	Relational Database Management System
RDS	Rapid Deployment Solutions
RFC	Remote Function Call
S/4HANA	New generation of Business suite
SAP HANA	High Performance Analytic Appliance from SAP
SAP NetWeaver	SAP's integrated technology computing platform
Sapadm	SAP Host Agent Administrator contains all required elements for centrally monitoring any host.
SCM	Supply Chain Management
Solution Manager	It provides central access to tools methods and preconfigured content that you can use during the evaluation, implementation, and productive operation of your systems.
SQLDBC	SQL Database Connectivity is a runtime library that enables applications to execute SQL statements in the database.

SRM	Stockholder Relationship Management
SUM	Software Update Manager
SUSE, Red Hat	Linux Distributions

Feedback

SAP SE has made every effort in the preparation of this course to ensure the accuracy and completeness of the materials. If you have any corrections or suggestions for improvement, please record them in the appropriate place in the course evaluation.