S02 - Native HANA Modeling - Part 1

SAP Native HANA Modeling Fundamentals	1. SDA Remote Source & HANA Virtual Tables & Staging tables 2. HANA Calculation Views 3. HANA Table functions 4. HANA DB Procedures 5. XS Job	2nd Aug 8:30 AM - 10:30 AM
Requirement	1. CV for Material Dimension 2. CV for Plant Dimension 3. CV for Sales Fact 4. CV combining Material, Plant and Sales Fact	2nd Aug 8:30 AM - 10:30 AM

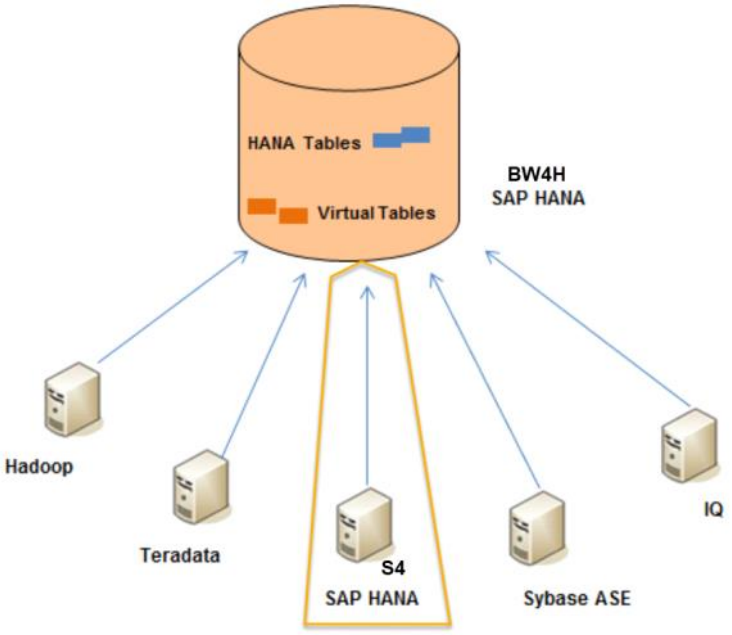
Q & A Session

What is Staging Table and Virtual Table? How to identify the table and how to see the data for that table?
What is Logic Cockpit and what is the uses of that?
What is the programming language used in HanaDB
What is the use of Outline tab in Eclpise
What is the backend for BW projects in eclipse? Is it BW or HanaDB!

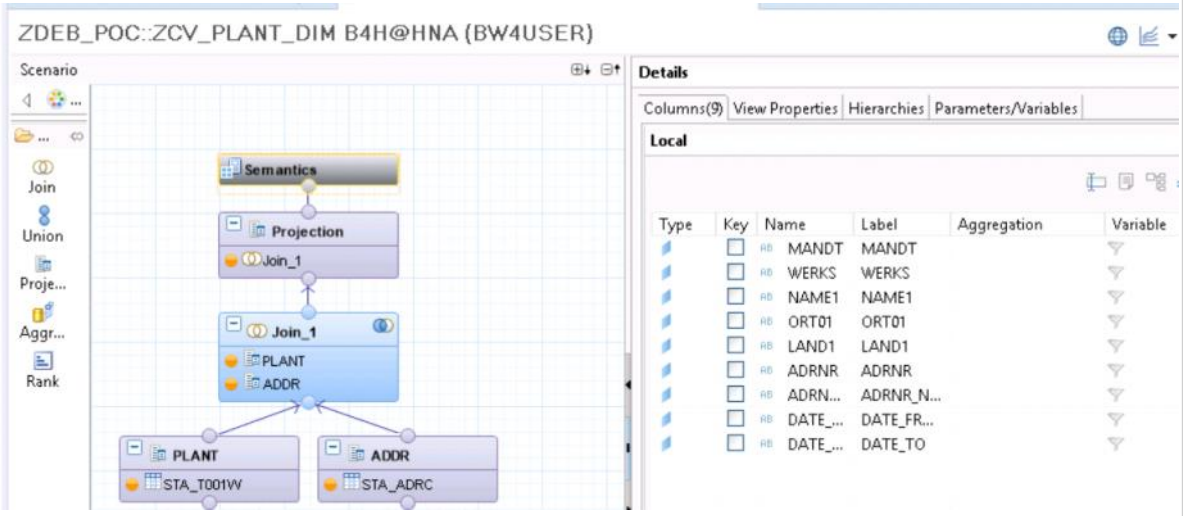
Answers:

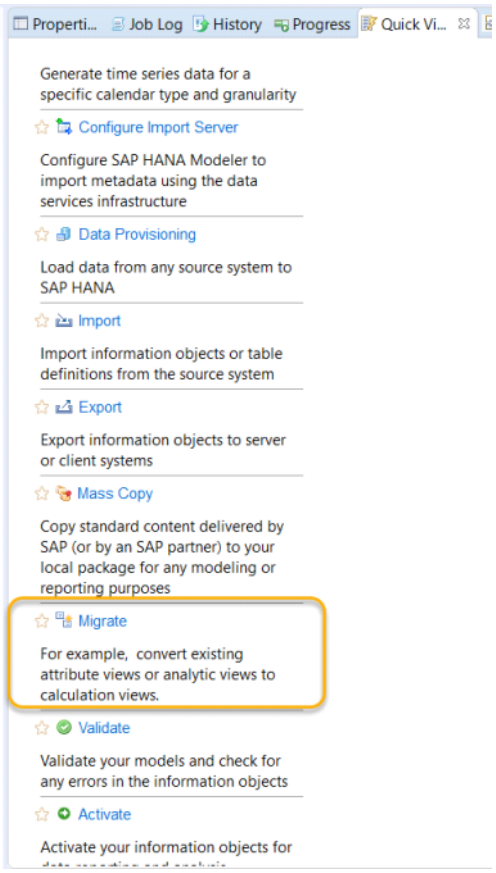
1. Virtual tables - fetch data in real time and they don't store any data.
   Stating tables - fetch data from local database, meaning they store data.
2. LO Cockpit - Set of logistics applications. Usage - activation of datasources.
3. SQL Script - Language used for modeling objects such as table functions, procedures etc.
   HANA also supports other languages such as JS, Java, Python etc.
4. Outline tab - Summarized info on the main object opened in the editor
5. Project backend is app server. Project is a connection to the app server.

SDA Remote Source

What is SDA	What is HANA Smart Data Access (SDA) ? SAP HANA smart data access enables remote data to be accessed. It enables remote data to be accessed as if they are local tables in SAP HANA, without copying the data into SAP HANA. Specifically, in SAP HANA, you can create virtual tables which point to remote tables in different data sources. Customers can then write SQL queries in SAP HANA, which could operate on virtual tables. Steps: 1. Install ODBC driver for the remote source DB. Lot of drivers given out of box. 2. Create Remote Source with a user which has all the required privileges in the remote source to access table/view data. 3. Create virtual table in target HANA DB. 4. Create persistent table in target HANA DB (optional, usually required for performance reasons). 5. Create CVs on top of either virtual table or persistent tables.
Our Scenario	

HANA CV - Calculation View

What is a HANA CV	A calculation view is a flexible information view that you can use to define more advanced slices on the data available in the SAP HANA database. Basically, a CV is a combination of tables/DB views/table functions and create a virtual model. E.g. We have material in one table. We have material text in another table. We have plant in one table. We have Sales Order item and quantity details in one table. A CV will combine all these tables via joins and will present one consolidated model. Calculation views are simple and yet powerful because they mirror the functionality found in both attribute views and analytic views, and also other analytic capabilities. 
Attribute Views	Deprecated. Replacement : HANA CV of type 'Dimension'. 
Analytic Views	Deprecated. Replacement : HANA CV of type 'Cube' or 'Cube with Star Join' 
Reason for Deprecation	Analytical views runs in: OLAP engine, Calculation views runs in: Calculation/SQL engine Attribute views runs in: Join engine. Table functions-based CV have replaced the SQL-Script-based CVs. As there is a lot of data transfers that may happen among all these engines and also some redundancy in their intrinsic features, the idea is to use only Calculation views running mostly in the SQL engine but you could still have it running in the Calculation engine while using Column engine "Conversion functions".
Migration Option in Eclipse	Via 'Quick View'

	
	MD CV for Material - Dimension CV for Plant - Dimension TD CV for Sales Order Item details - Cube MD + TD CV for MD + TD - Cube with Star Join
	USER/SCHEMA: _SYS_REPO

System vs Repo

System	Repo
No CICO	CICO
No repo SP and no repo TF	Repo SP Repo TF
No repo Table definition	Repo table definitions
Easy copy CVs	Cannot copy CVs
Easy rename CVs	No renaming CVs

App Server: BW4HANA 2.0 SP02 or above
DB: HANA 2.0 SP4
ODP_CDS Source System
SDA
HANA Local Schema SS
HANA SDA SS

System repository user for HANA objects

Backend repository owner	_SYS_REPO	Knows about the schema and tables

HANA Join Types

HANA Join types