

Kapitel 3

Einführung in die SAP-HANA-Datenmodellierung

In diesem Kapitel führen wir Sie in die native SAP-HANA-Modellierung ein. Dabei liegt der Schwerpunkt auf der Erstellung von Information Views auf SAP-HANA-Datenbankobjekte, um die Berichtsanforderungen zu erfüllen.

Es gibt drei Hauptansätze für den Einsatz von SAP HANA als Grundlage für die Entwicklung von Datenflüssen (für das Reporting):

Hauptansätze für den Einsatz von SAP HANA

- **Native SAP-HANA-Modellierung**
Bei diesem Ansatz geht es um die Entwicklung von *Information Views*, auch Informationsmodelle genannt. Diese beziehen sich auf Tabellen und andere SAP-HANA-Datenbankobjekte (z. B. CDS Views).

Die native Modellierung wird vollständig mit Hilfe der Funktionen und Werkzeuge durchgeführt, die SAP HANA zur Verfügung stellt. Die Informationen aus den Tabellen werden dabei mit denen anderer Datenbankobjekte kombiniert, um den Anforderungen des Berichtswesens zu entsprechen.

Die native Modellierung kommt zum Einsatz, wenn eine Anwendung auf der SAP-HANA-Datenbank betrieben wird und man direkt die Daten abfragen möchte, die in den Datenbanktabellen erfasst werden. Es ist aber auch möglich, auf Remote-Datenquellen zuzugreifen und auf deren Basis Information Views zu erstellen.

Bei den Information Views unterscheiden wir zwischen Calculation Views, Analytic Views und Attribute Views.

- **Verwendung von SAP BW bzw. SAP BW/4HANA**
Wenn Sie ein SAP-BW-System auf SAP HANA betreiben oder SAP BW/4HANA einsetzen, erfolgt die Datenmodellierung in einer Eclipse-basierten, integrierten Entwicklungsumgebung (in SAP HANA Studio bzw. den BW-Modellierungswerkzeugen) und zum Teil auch noch in der SAP-GUI-basierten Workbench in SAP BW.

SAP HANA wird in diesem Szenario als Datenbank zur Speicherung der Datenbankobjekte (Tabellen und Views) verwendet, die bei der Model-

lierung in der SAP-GUI-Workbench- und Eclipse-Umgebung erzeugt werden.

■ Gemischtes/hybrides Szenario

Es gibt auch das gemischte Szenario, in dem die beiden genannten Methoden kombiniert werden. Beim Anlegen eines BW-Modellierungsobjekts (d. h. eines InfoObjects, advanced DataStore-Objekts oder CompositeProviders) kann man veranlassen, dass automatisch Calculation Views erzeugt werden. Diese automatisch erzeugten Calculation Views kann man in der nativen SAP-HANA-Modellierung weiterverwenden.

In diesem Kapitel konzentrieren wir uns auf die native SAP-HANA-Modellierung, also die Erstellung von Informationsmodellen. Dazu werden Sie zunächst die Umgebung von SAP HANA Studio kennenlernen. Sie erfahren außerdem, welche Arten von Information Views es gibt. Anschließend beschäftigen wir uns damit, wie Sie einfache Information Views anlegen. Anspruchsvollere Modellierungsaufgaben behandeln wir dann im folgenden Kapitel.

3.1 SAP HANA Studio

Bevor Sie jedoch Information Views in SAP HANA Studio erstellen können, müssen Sie es installieren und mit der SAP-HANA-Datenbank verbinden.

Installation Zunächst müssen Sie den SAP HANA Studio Client in Eclipse installieren.



Installationsanleitungen von SAP

Wie Sie bei der Installation vorgehen, erfahren Sie auf den folgenden Seiten von SAP:

- Eclipse: <https://tools.hana.ondemand.com/>
- SAP HANA Studio: <https://help.sap.com/viewer/a2a49126a5c546a9864aae22c05c3d0e/2.0.02/en-US>

Perspektiven SAP HANA Studio bietet verschiedene *Perspektiven*, mit denen man in Eclipse arbeiten kann. Die Perspektiven beinhalten jeweils verschiedene Funktionen, die es Ihnen ermöglichen, mit dem SAP-HANA-System zu interagieren.

In Abbildung 3.1 sehen Sie einige der verfügbaren Perspektiven in SAP HANA Studio:



Abbildung 3.1 Perspektiven in SAP HANA Studio

- Um z. B. BI-Lösungen (Business Intelligence) mit SAP BW auf SAP HANA zu entwerfen und zu implementieren, verwenden Sie die Perspektive **BW Modeling**, d. h. die BW-Modellierungswerkzeuge in Eclipse.
- Um Systemadministrationsaufgaben auszuführen, müssen Sie sich in der Perspektive der **SAP HANA Administration Console** befinden.
- Die native SAP-HANA-Modellierung wird größtenteils in der Perspektive **SAP HANA Modeler** von BI-Experten ausgeführt. Dies ist auch die Perspektive, die Sie im Folgenden verwenden, um eine Verbindung mit der SAP-HANA-Datenbank herzustellen.

Im Folgenden zeigen wir Ihnen, wie Sie die Perspektive SAP HANA Modeler in SAP HANA Studio öffnen, um von dort aus eine Verbindung zum SAP-HANA-System herzustellen.

1. Wählen Sie dazu den folgenden Menüpfad in SAP HANA Studio:
Window • Perspective • Open Perspective • SAP HANA Modeler
2. Nun klicken Sie auf das Symbol **Add System** (🔌), um ein neues System anzubinden, wie in Abbildung 3.2 gezeigt.

Verbindung zur SAP-HANA-Datenbank

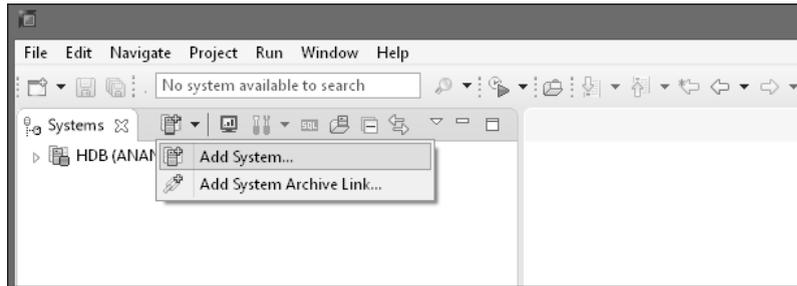


Abbildung 3.2 System in SAP HANA Studio hinzufügen

3. Geben Sie als Nächstes die Details zur Verbindung mit dem SAP-HANA-System an (siehe Abbildung 3.3). Sie können diese Systeminformationen bei Ihrem System- oder Datenbankadministrator erfragen.

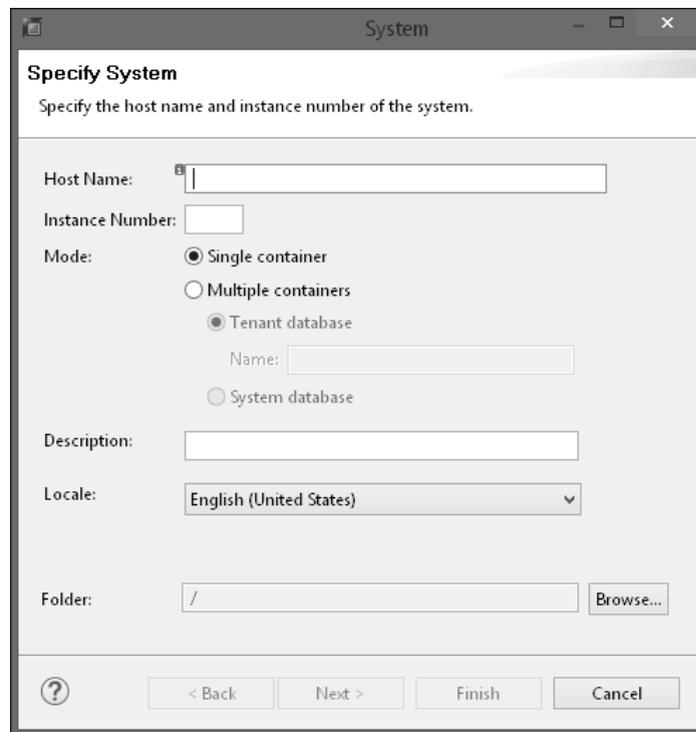


Abbildung 3.3 Systemdetails angeben

4. Nachdem Sie die erforderlichen Angaben gemacht haben, klicken Sie auf **Next**. Als Nächstes geben Sie den Benutzernamen und das Passwort für das SAP-HANA-System ein und klicken abschließend auf **Finish** (siehe Abbildung 3.4).

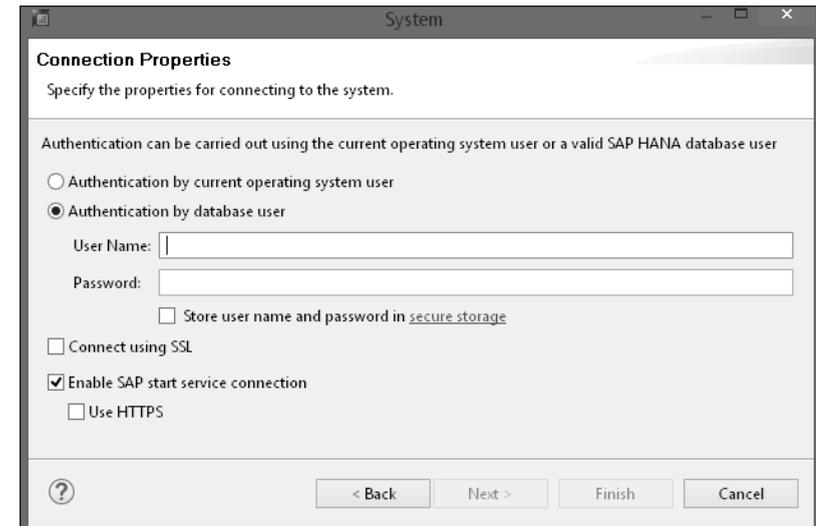


Abbildung 3.4 Benutzer-Login-Daten angeben

Nach diesem Schritt haben Sie eine Verbindung zur SAP-HANA-Datenbank erstellt.

In der Perspektive SAP HANA Modeler werden Ihnen nun unterhalb dieser Datenbank vier Ordner angezeigt (siehe Abbildung 3.5), die die verschiedenen Aspekte der SAP-HANA-Umgebung aus Modellierungssicht abbilden.

Ordner für
Modellierungs-
objekte

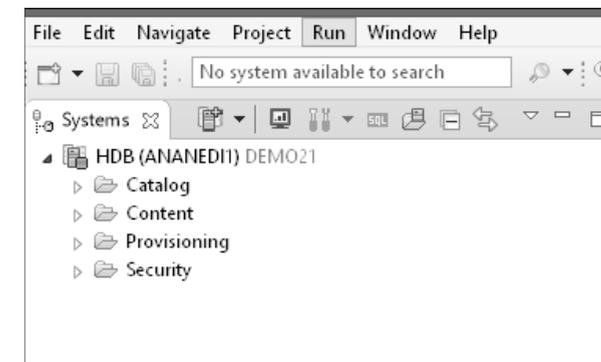


Abbildung 3.5 Ordner für SAP-HANA-Datenbankobjekte

In diesen Ordnern werden die modellierungsrelevanten Informationen gespeichert:

- **Catalog**
Bietet Zugriff auf alle im System definierten Schemata, einschließlich System- und Benutzerschemata.

- **Provisioning**
Dient der Verbindung zu anderen Remote-Quellen. Hier wird der Smart Data Access konfiguriert.
- **Content**
Der Content-Ordner ist in Pakete (**Packages**) unterteilt. In diesen Paketen werden die Information Views organisiert.
- **Security**
Hier werden die Details zu Benutzerautorisierung, Rollen und Sicherheitsberechtigungen definiert.

Im Folgenden sehen wir uns die Inhalte der einzelnen Ordner genauer an.

3.1.1 Catalog-Ordner

Der Catalog besteht aus Schemata. Ein *Schema* ist ein Container, der Datenbankobjekte wie Tabellen, Ansichten (Views) und Datenbankprozeduren (Stored Procedures) enthält. In Abbildung 3.6 sind einige Schemata dargestellt.

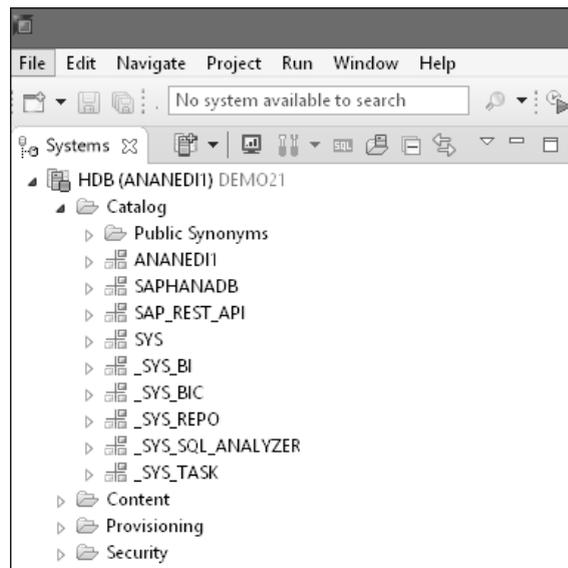


Abbildung 3.6 Schemata im Catalog-Ordner

Arten von Schemata Es gibt zwei verschiedene Arten von Schemata: system- und benutzerspezifische Schemata:

- Wenn ein Benutzer im SAP-HANA-System angelegt wird, wird ein Schema erstellt, in dem dieser Benutzer seine Tabellen, Ansichten usw. spei-

chern kann. Ein Benutzer kann auch zusätzliche Schemata erstellen (wenn er dazu berechtigt ist).

- Systemschemata werden automatisch erstellt. Ein Beispiel für ein solches Schema ist das Schema, in dem SAP HANA die SAP-Anwendungstabellen speichert.

Um ein Schema zu erstellen, gehen Sie wie folgt vor:

Schema erstellen

1. Öffnen Sie die SQL-Konsole über die entsprechende Schaltfläche (SQL) in der Funktionsleiste der Registerkarte **Systems** (siehe Abbildung 3.7).

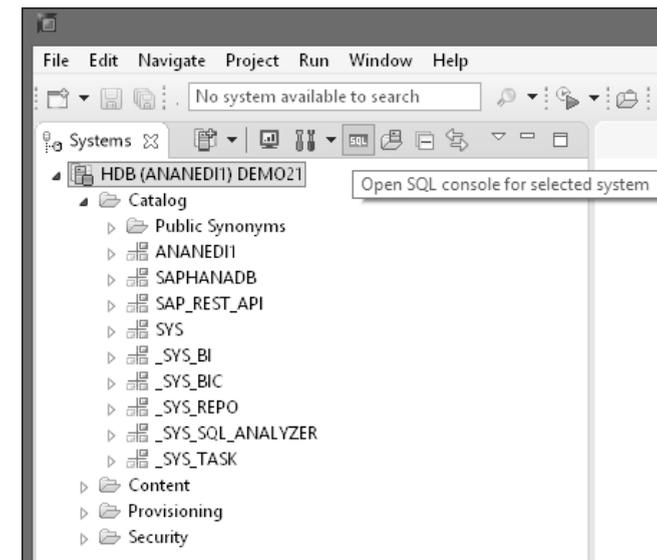


Abbildung 3.7 SQL-Konsole öffnen

2. Geben Sie in der SQL-Konsole den folgenden Befehl ein (siehe Abbildung 3.8): `CREATE SCHEMA TEST`. Damit erstellen Sie ein Schema namens TEST.

Jetzt können Sie Tabellen in diesem Schema erstellen.

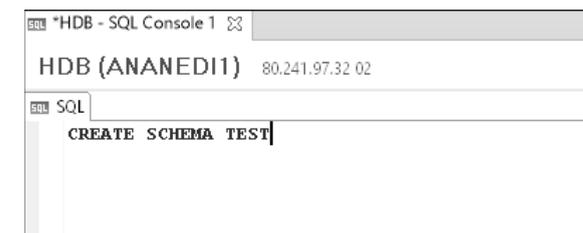


Abbildung 3.8 SQL Statement in SQL Console

Schema finden In einer Umgebung, in der SAP HANA z. B. als Datenbank für SAP-Transaktionsanwendungen verwendet wird, werden die von den Anwendungen verwendeten Tabellen in einem bestimmten Systemschema gespeichert. Um den Namen dieses Schemas zu finden, melden Sie sich über SAP GUI an und navigieren zu **System • Status**. Sehen Sie sich hier den Bereich **Database data** an. In Abbildung 3.9 heißt das Schema z. B. **SAPHANADB**.

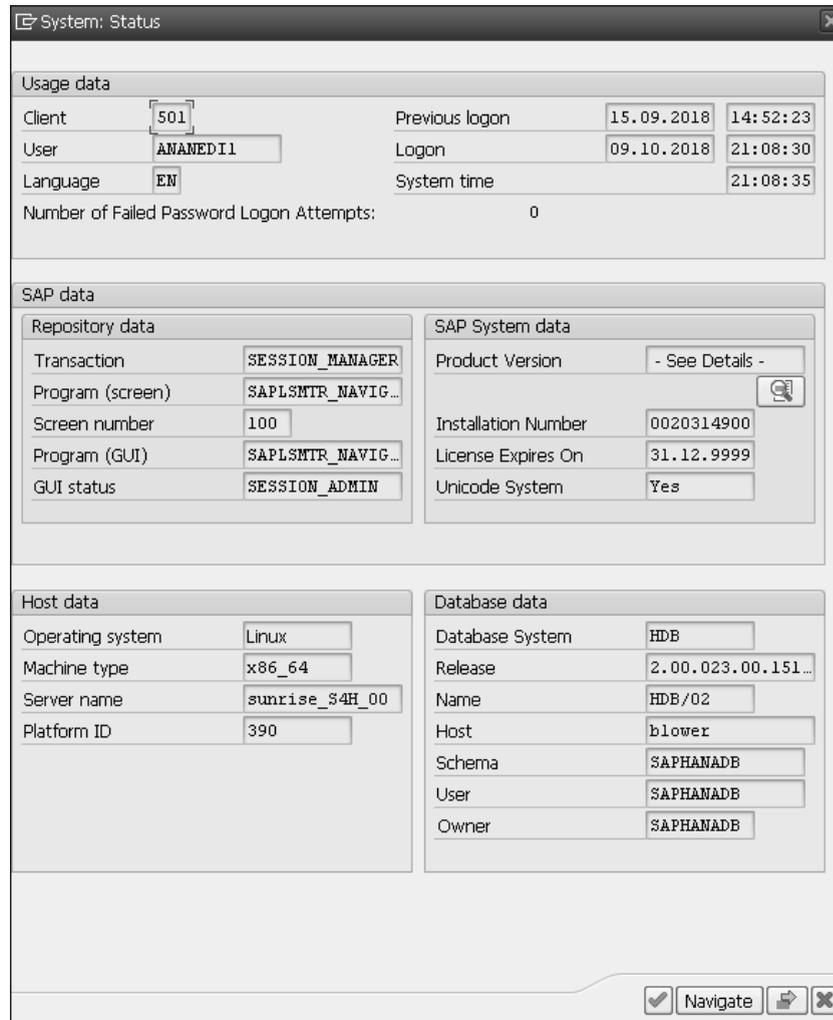


Abbildung 3.9 Schema für eine SAP-Anwendung heraussuchen

Anwendungstabellen anzeigen Das gleiche Schema finden Sie auch in SAP HANA Studio (siehe Abbildung 3.10). Wenn Sie den Ordner **Tables** öffnen, werden Ihnen im rechten Fenster die Tabellen angezeigt, die sich in diesem Schema befinden.

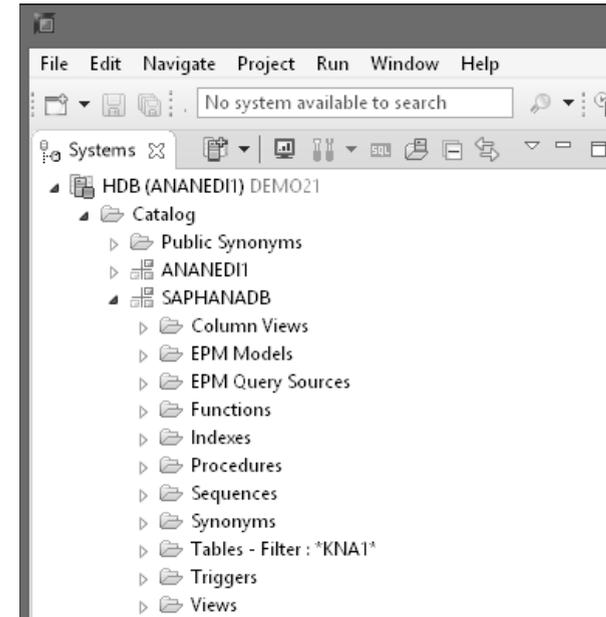


Abbildung 3.10 SAP-Anwendungsschema in der SAP-HANA-Datenbank

In Abbildung 3.11 werden z. B. die Metadaten der Tabelle KNA1 im Schema SAPHANADB angezeigt.

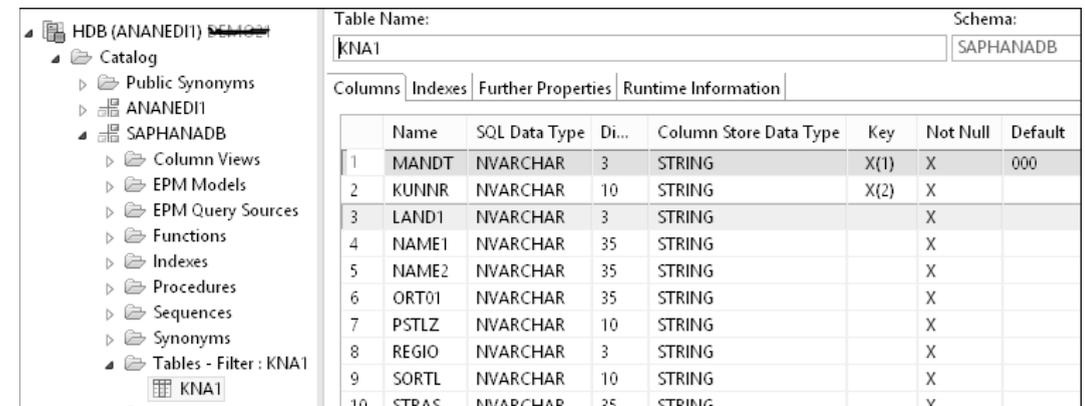


Abbildung 3.11 Metadaten der Tabelle KNA1 (Kundendaten) im Schema SAPHANADB

Sie können diese Tabelle aufrufen, indem Sie die Funktion **Find Table** (siehe Abbildung 3.12) im Kontextmenü des Ordners **Table** nutzen und gezielt nach dem Namen der Tabelle suchen. Alternativ können Sie über die Funktion **Filters** einen Filter auf die Tabelle KNA1 setzen.

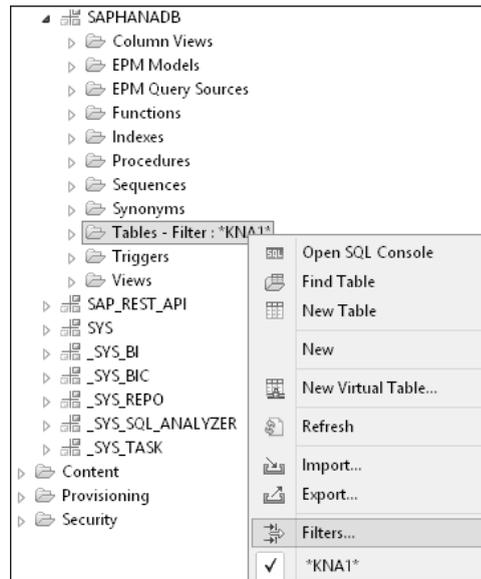


Abbildung 3.12 Tabelle suchen oder danach filtern

Weitere System-schemata

Die anderen Systemschemata, die für die Datenmodellierung von Interesse sind, sind:

- **__SYS_BI**
Dieses Schema enthält z. B. die Tabellen für die Kalenderdaten (fiskal, gregorianisch), die zum Generieren von Zeitdimensionen verwendet werden. Zeitdimension werden für multidimensionale Datenanalysen verwendet.
- **__SYS_BIC**
Wenn der Benutzer einen Information View (Attribute View, Analytic View, Calculation View) ein Analytic Privilege oder eine Stored Procedure aktiviert, werden die entsprechenden Laufzeitobjekte unter dem Schema __SYS_BIC erstellt. Dieses Schema enthält alle Column Views der aktivierten Objekte.
- **__SYS_REPO**
__SYS_REPO ist ein Systembenutzer, der von SAP HANA zu Aktivierungszwecken verwendet wird. Sein Schema ist ein Systemschema und trägt denselben Namen __SYS_REPO.
Wenn Sie ein neues Anwendungsschema anlegen und unter diesem Schema Informationsmodelle erstellen und aktivieren möchten, müssen Sie diesem Benutzer eine Berechtigung für Ihr Schema gewähren. Er erstellt die erforderlichen Laufzeitobjekte aus Ihren Modellen und speichert sie im Schema __SYS_BIC.

Nachdem Sie Ihre Modelle aktiviert haben, wird __SYS_REPO erneut verwendet, um Ihnen (und wahrscheinlich anderen Benutzern) Lesezugriff auf die aktivierten Modelle zu geben. Die Berechtigung für Ihr Schema können Sie dem Benutzer __SYS_REPO mit der folgenden SQL-Anweisung zuweisen:

```
GRANT SELECT ON SCHEMA<SCHEMA_NAME> TO __SYS_REPO
WITH GRANT OPTION
```

3.1.2 Content-Ordner

Der Content-Ordner besteht aus Paketen. Diese werden von Benutzern erstellt und dienen der Speicherung der Information Views, also der Datenmodelle. Um ein Datenmodell zu erstellen, müssen Sie daher zunächst ein Paket im Content-Ordner anlegen. Pakete können wiederum Unterpakete enthalten.

Pakete

Um ein Paket anzulegen, wählen Sie im Kontextmenü des Content-Ordners **New • Package**. Geben Sie dann einen Namen und eine Beschreibung für das Paket ein (siehe Abbildung 3.13). Wenn Sie ein Paket unterhalb eines existierenden Pakets anlegen möchten, wählen Sie im Kontextmenü dieses Pakets **New • Package**.

Paket anlegen

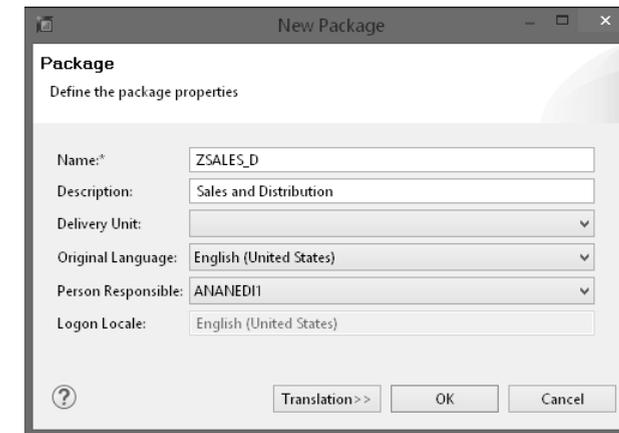


Abbildung 3.13 Neues Paket erstellen

Die Datenmodelle (also die Attribute, Analytic und Calculation Views usw.) werden unterhalb eines Pakets erstellt und dort organisiert. Die Pakete können daher mit einer InfoArea in SAP BW verglichen und als solche nach Themenbereichen gegliedert werden. Für die Pakete können Berechtigungen eingerichtet werden, um sicherzustellen, dass nur autorisierte Benutzer Zugriff auf den Inhalt eines Pakets haben.

Delivery Units Pakete sind Liefereinheiten (*Delivery Units*) zugeordnet. Eine Liefereinheit wird verwendet, um Pakete so zu gruppieren, dass sie von einem SAP-HANA-System zu einem anderen transportiert werden können. Wenn Sie ein Paket mit dem Ziel erstellen, es zu einem anderen SAP-HANA-System zu transportieren (z. B. vom Entwicklungssystem zum Qualitätssicherungssystem), müssen Sie dem Paket eine Liefereinheit zuordnen.

3.1.3 Provisioning-Ordner

Der Provisioning-Ordner wird zum Konfigurieren von Remote-Quellen und für den Einsatz von SAP HANA Smart Data Access (SDA) verwendet. Mit Hilfe von SDA kann SAP HANA auf Tabellen in entfernten Datenquellen zugreifen. Auf diese Weise können Sie ein verteiltes Informationssystem implementieren, indem Sie die Daten nicht physisch aus dem Quellsystem in SAP HANA laden. Die Daten bleiben im Quellsystem. Sie importieren die Metadaten der Tabellen aus dem Quellsystem in SAP HANA und erstellen dann virtuelle Datenbanktabellen für diese Tabellen. Für diese virtuellen Tabellen können Sie dann Information Views bzw. Datenmodelle für das Berichtswesen erstellen.

Adapter für den Remote-Datenzugriff

Damit SAP HANA mit den entfernten Quellsystemen kommunizieren kann, müssen für jede Art von Quellsystem bestimmte Adapter zur Verfügung stehen. Wenn Sie versuchen, in SAP HANA eine Remote-Verbindung einzurichten, sehen Sie die Liste der Adapter, die Ihnen zur Verfügung stehen (siehe Abbildung 3.14). Um eine Remote-Verbindung einzurichten, öffnen Sie das Kontextmenü des Ordners **Remote Sources** unterhalb des Ordners Ihrer SAP-HANA-Datenbank und wählen **New • Remote Source**.

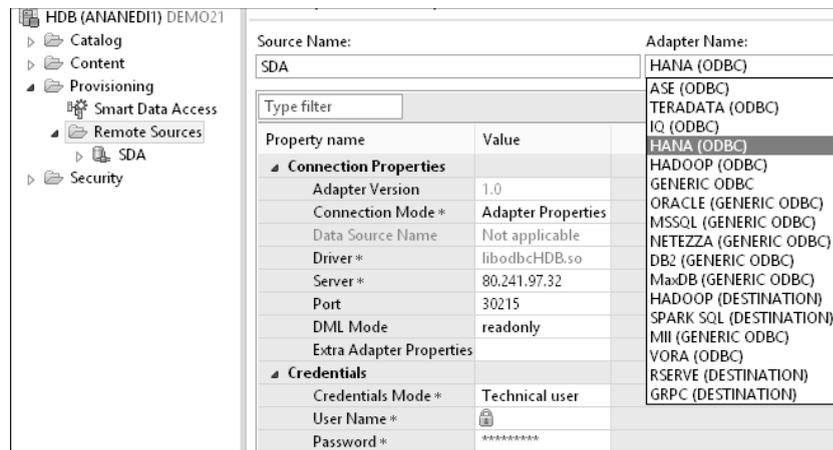


Abbildung 3.14 Adapter für Remote-Datenzugriff

3.1.4 Security-Ordner

In dem Ordner **Security** werden die Benutzer des SAP-HANA-Systems und ihre Rollen angelegt und verwaltet (siehe Abbildung 3.15). Die Rollen und Berechtigungen (im Umfeld von SAP HANA spricht man von *Privilegien*), die einem Benutzer zugewiesen wurden, bestimmen die Aktionen, die dieser Benutzer im System ausführen kann.

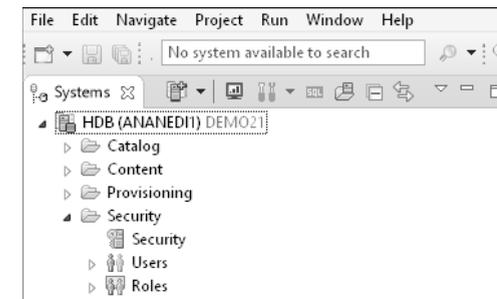


Abbildung 3.15 Inhalt des Security-Ordners

Klicken Sie auf den Ordner **Users**, um in die Benutzerverwaltung zu gelangen. Abbildung 3.16 zeigt ein Beispiel für einen Benutzer. Im oberen Bereich wird die Authentifizierungsmethode definiert. Unten sehen Sie die diesem Benutzer zugewiesenen Berechtigungen.

Benutzerverwaltung

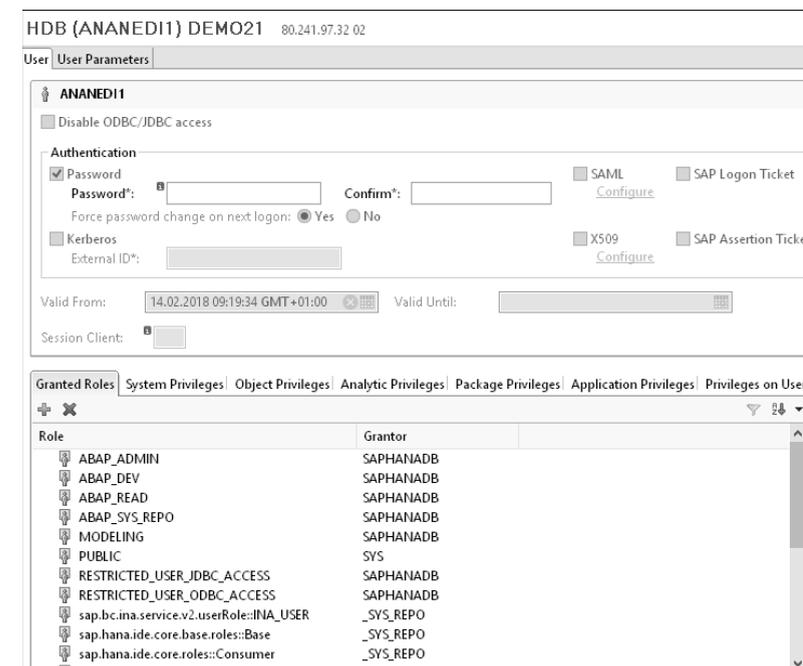


Abbildung 3.16 Berechtigungen, die einem Benutzer zugewiesen sind

Nachdem Sie nun die verschiedenen Ordner der SAP-HANA-Umgebung kennengelernt haben, zeigen wir Ihnen als Nächstes, wie Sie Information Views erstellen.

3.2 Information Views

Modellierungs- optionen

Ziel der nativen SAP-HANA-Modellierung ist es, die Daten in den Tabellen der SAP-HANA-Datenbank so bereitzustellen, dass sie die Berichtsanforderungen eines Unternehmens erfüllen. Hierbei werden die Tabellen z. B. verknüpft (Join) oder zusammengeführt (Union). Dadurch erstellen Sie Information Views (Informationsmodelle). Berichterstellungswerkzeuge wie SAP Lumira, SAP Analysis for Microsoft Office usw. können auf diese Information Views zugreifen und Daten anzeigen. Um Information Views zu erstellen, werden aber nicht nur Tabellen genutzt, sondern, wenn erforderlich, auch CDS-(Core Data Services-)Entitäten, generierte externe Views aus SAP BW bzw. SAP BW/4HANA und SQL Views.

Bei der Modellierung von Information Views können Sie abhängig von den Berichtsanforderungen entscheiden, ob Sie die gesamte Tabelle oder nur eine Teilmenge der Daten aus den zugrundeliegenden Tabellen brauchen. Aus den ursprünglichen Attributen und Kennzahlen der Tabellen können Sie auch neue Attribute und Kennzahlen ableiten und berechnen. Diese neuen abgeleiteten oder berechneten Attribute und Kennzahlen sind virtuell und befinden sich nicht physisch in den Datenbanktabellen. Mit Hilfe von Information Views werden dadurch virtuelle Modelle erstellt. Diese virtuellen Modelle werden in den Paketen im Content-Ordner angelegt und verwaltet.

Information View anlegen

Um einen neuen Information View anzulegen, öffnen Sie das Kontextmenü zu einem Paket und wählen dort **New** und den View-Typ (siehe Abbildung 3.17).

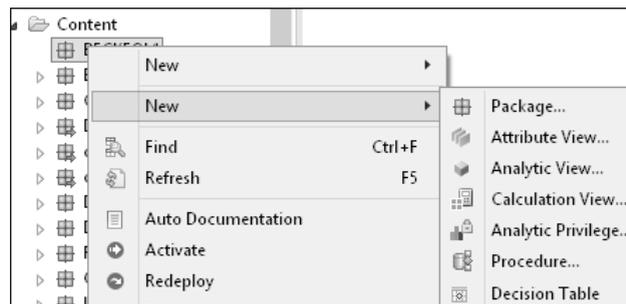


Abbildung 3.17 Information View werden unterhalb der Pakete erstellt.

Es gibt verschiedene Information-View-Typen, aus denen Sie bei der Anlage eines Information Views auswählen. Diese sind:

Information-View- Typen

- Attribute Views
- Analytic Views
- Calculation Views

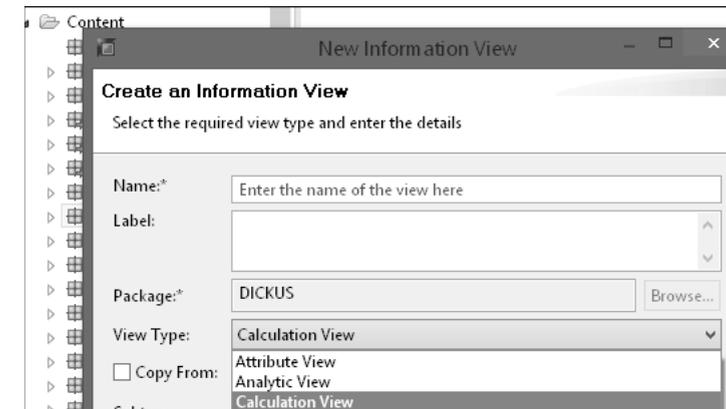


Abbildung 3.18 Verschiedene Information-View-Typen

Um Information Views auszuführen, verwendet SAP HANA die sogenannten SAP HANA Engines. Es gibt die folgenden Engines:

SAP HANA Engines

- Join Engine
- OLAP Engine
- Calculation Engine

Während der Ausführung werden Optimierungsprozesse durchgeführt, die sicherstellen, dass die geeignete Engine verwendet wird.

Die View-Typen Attribute und Analytic View sind inzwischen veraltet. Sie wurden für folgende Zwecke verwendet:

Attribute und Analytic Views

- *Attribute Views* wurden für die Modellierung von Stammdaten oder Dimensionsanforderungen verwendet. Dabei wurden Views über Tabellen erstellt, die Stammdateninformationen enthalten, z. B. Kunden, Materialien, Produkte, Mitarbeiter usw. Der Fokus lag hier auf der Modellierung von Attributen und nicht auf den Kennzahlen.
- *Analytic Views* wurden für die Modellierung von Sternschemata verwendet (siehe auch Abschnitt 1.5.2, »Sternschema«). Ein Sternschema besteht aus einer Faktentabelle (Bewegungsdaten mit Kennzahlen), die von Dimensionen (Stammdaten) umgeben ist. Um einen Analytic View zu modellieren, benötigte man daher einige Attribute Views, die wie in Ab-

Abbildung 3.19 zeigt mit einer Faktentabelle verknüpft sind. Analytic Views können grob mit den InfoCubes in einem klassischen SAP-BW-System verglichen werden.

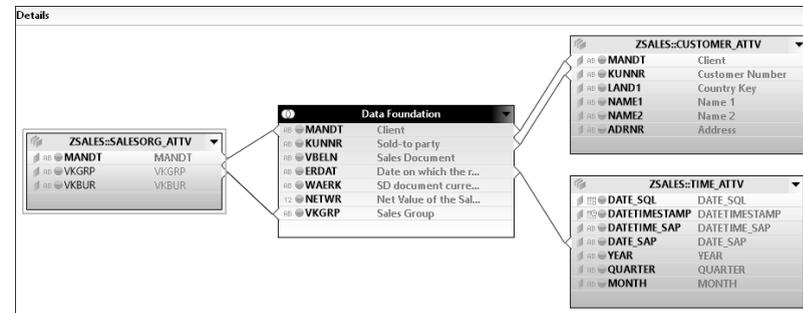


Abbildung 3.19 Grafische Modellierung eines Analytic View (Sternschema)

Obwohl Attribute und Analytic View veraltet sind, werden sie in den aktuellen Versionen von SAP HANA weiterhin unterstützt.

Calculation View

Bei den neuesten Versionen von SAP HANA empfiehlt es sich jedoch, *Calculation Views* zu verwenden. Calculation Views bieten die größte Flexibilität bei der Datenmodellierung. Man unterscheidet hier zwischen zwei Untertypen (siehe Abbildung 3.20):

- **Standard**
Standard-Calculation-Views dienen zur Modellierung von Stamm- und Bewegungsdaten (z. B. Kundenaufträgen, Kundendaten usw.).
- **Time**
Zeitbasierte Calculation Views dienen zur Modellierung zeitspezifischer Daten (z. B. Jahr, Monat, Tage usw. oder Steuerdaten).

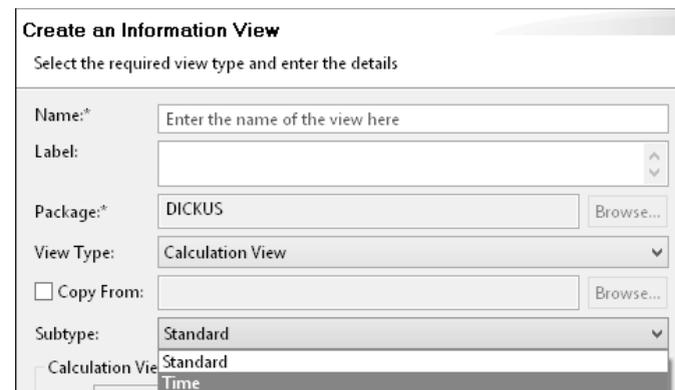


Abbildung 3.20 Untertypen von Calculation Views: »Standard« oder »Time«

Zeitbasierte Calculation Views können mit Standard-Calculation-Views verknüpft werden, um zeitbezogene Datenanalysen durchzuführen.

Standard-Calculation-Views können entweder grafisch oder mit Hilfe von SQLScript erstellt werden (siehe Abbildung 3.21). Mit dem grafischen Ansatz können Sie auch modellieren, wenn Sie keine vertieften Kenntnisse der Sprache SQL haben. Für den SQLScript-Ansatz sind dagegen SQL-Kenntnisse erforderlich. Der SQLScript-Ansatz ist erforderlich, wenn die grafischen Calculation Views die Berichtsanforderungen nicht erfüllen können, insbesondere in Fällen, in denen komplexe Programmierlogik benötigt wird. Es wird in den aktuellen SAP-HANA-Releases jedoch nicht mehr empfohlen, die Option **SQL Script** an dieser Stelle auszuwählen, sondern direkt Tabellenfunktionen oder Datenbankprozeduren in der Perspektive **SAP HANA Development** zu entwickeln. Wie Sie bei dieser codebasierten Modellierung vorgehen, erklären wir in Abschnitt 4.9, »Codebasierte Modellierung«.

Standard-Calculation-Views

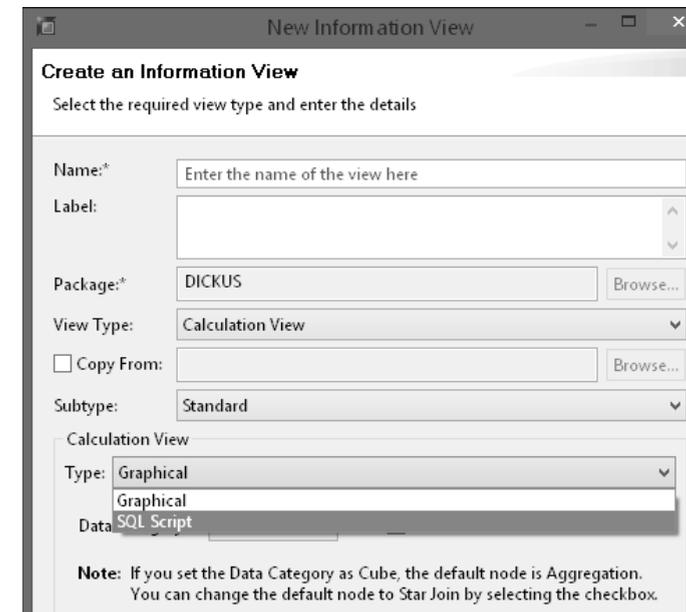


Abbildung 3.21 Calculation-View-Typen: »Graphical« oder »SQL Script«

Beim Erstellen von Standard-Calculation-Views müssen Sie außerdem zwischen den folgenden Datenkategorien (**Data category**) wählen (siehe Abbildung 3.22):

Datenkategorien

- **Dimension Calculation View**
Calculation Views dieser Kategorie sind den Attribute Views ähnlich und werden für die Modellierung von Stammdaten verwendet (z. B. Kunden, Materialien, Produkte, Mitarbeiter usw.). Dimension Calculation Views

werden normalerweise in anderen Calculation Views, etwa einem Cube Calculation View mit Star Join) für mehrdimensionale Berichtszwecke verwendet. In diesem Fall werden die für die Analyse erforderlichen Stammdateninformationen durch die Dimension Calculation Views bereitgestellt.

■ Cube Calculation View

Diese Art von Calculation View wird zur Modellierung von Bewegungsdaten für das Reporting verwendet. Wenn Sie z. B. Kundenaufträge analysieren möchten, können Sie einen Cube Calculation View verwenden, um diesen Zweck zu erfüllen. In dieser Hinsicht sind die Cube Calculation Views vergleichbar mit einem Standard-DataStore-Objekt (DSO) in SAP BW.

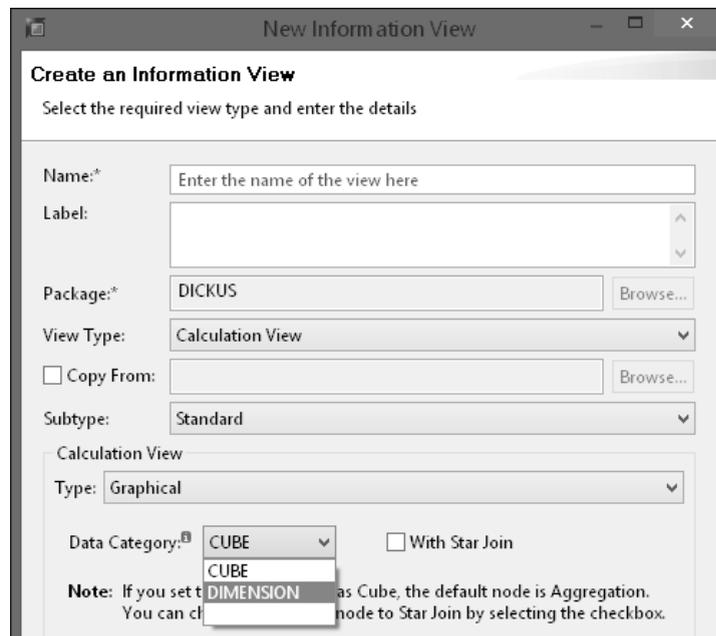


Abbildung 3.22 Calculation View: Unterschiedliche Data-Category-Optionen

■ Cube Calculation View (mit Star Join)

Wählen Sie die Option **With Star Join**, ähnelt der Cube Calculation View einem Sternschema für mehrdimensionale Datenanalysen und Berichterstellung. Er besteht aus einer Faktentabelle, die Kennzahlen enthält und von Dimension Calculation Views umgeben ist. Ähnlich wie der Analytic View ist der Cube Calculation View mit Star Join also vergleichbar mit einem InfoCube in SAP BW. Wie Sie bei der Modellierung eines Calculation Views vorgehen, erfahren Sie in Abschnitt 3.3.2, »Dimension Calculation Views«.

3.3 Einfache Calculation Views modellieren

In diesem Abschnitt zeigen wir Ihnen, wie Sie Ihre ersten grafischen Calculation Views modellieren. Dazu klären wir zunächst, welche Datenquellen Sie bei der Modellierung verwenden können. Anschließend zeigen wir Ihnen, wie Sie einen Dimension Calculation View anlegen und mit einem Text Join erweitern. Modellierungsbeispiele für Cube Calculation Views und weitere Aufgaben finden Sie in Kapitel 4, »Fortgeschrittene SAP-HANA-Datenmodellierung«.

3.3.1 Datenquellen grafischer Calculation Views

Die folgende Liste führt die wichtigsten Datenquellentypen auf, die beim Erstellen von Calculation Views verwendet werden können:

Datenquellentypen

- CDS Views
- SQL Views
- weitere Calculation Views
- Tabellenfunktionen
- Entscheidungstabellen
- virtuelle Tabellen (über SAP HANA Smart Data Access)
- Analytic Views
- SAP-HANA-Tabellen

In SAP HANA gibt es zwei Arten von Tabellen: zeilen- und spaltenbasierte Tabellen.

Row und Column Store

- In zeilenbasierten Tabellen (*Row Tables*) werden Tabellendatensätze in einer Sequenz von Zeilen gespeichert (*Row Store*).
- In spaltenbasierten Tabellen (*Column Tables*) werden Tabellendatensätze in einer Sequenz von Spalten gespeichert, d. h. die Einträge einer Spalte werden an zusammenhängenden Speicherplätzen gespeichert (*Column Store*).

Bevor Sie Ihre ersten Calculation Views erstellen, zeigen wir Ihnen als Hintergrund, wie Daten in SAP HANA organisiert werden. Beim Modellieren von Information Views nutzen Sie in der Regel SAP-Standardtabellen. Es kann aber auch vorkommen, dass Sie Ihre eigenen Tabellen anlegen möchten. Wenn Sie eine Tabelle in SAP HANA anlegen, müssen Sie angeben, ob es sich um eine spalten- oder zeilenbasierte Tabelle handelt. Standardmäßig wird eine zeilenbasierte Tabelle erstellt. Wenn Sie eine spaltenbasierte

Tabellen anlegen

Tabelle anlegen möchten, müssen Sie daher explizit den Typ der Tabelle angeben.

Datenorganisation

Anhand der folgenden Tabellen zeigen wir, wie Daten in SAP HANA organisiert sind. Eine Tabelle mit Umsatzdaten (siehe Tabelle 3.1) wird in SAP HANA unterschiedlich organisiert, je nachdem, ob sie als zeilenbasierte Tabelle (siehe Tabelle 3.2) oder als spaltenbasierte Tabelle (siehe Tabelle 3.3) angelegt wurde.

Land	Produkt	Umsatz
England	Kaffee	1.000
England	Tee	2.000
Deutschland	Cola	400
USA	Limonade	500

Tabelle 3.1 Umsatztabelle

Zeile 1	England
	Kaffee
	1.000
Zeile 2	England
	Tee
	2.000
Zeile 3	Deutschland
	Cola
	400
Zeile 4	USA
	Limonade
	500

Tabelle 3.2 Zeilenbasierte Tabelle

Spalte »Land«	England
	England
	Deutschland
	USA
Spalte »Produkt«	Kaffee
	Tee
	Cola
	Minonade
Spalte »Umsatz«	1.000
	2.000
	400
	500

Tabelle 3.3 Spaltenbasierte Tabelle

Um eine spaltenbasierte Tabelle zu erstellen, verwenden Sie z. B. die SQL-Anweisungen aus Listing 3.1:

```
CREATE COLUMN TABLE Kunde(ID INT,
    NAME VARCHAR(20),
    NACHNAME VARCHAR(30),
    PRIMARY KEY(ID));
```

Listing 3.1 SQL-Anweisung zum Anlegen einer Tabelle

Der Typ einer Tabelle (**Column Store** oder **Row Store**) wird Ihnen angezeigt, wenn Sie diese Tabelle unter **Catalog • Tables** in SAP HANA Studio anzeigen lassen. Abbildung 3.23 zeigt eine Tabelle vom Typ **Column Store**.

Table Name:		Schema:	Type:
KUNDE		ANANEDI1	Column Store
Columns			
Name	SQL Data Type	Di...	Column Store Data Type
1 ID	INTEGER		INT
2 NAME	VARCHAR	20	STRING
3 NACHNAME	VARCHAR	30	STRING

Abbildung 3.23 Anzeige des Tabellentyps

Auf spaltenbasierte Tabellen kann für die Datenanalyse schneller zugegriffen werden. Zeilenbasierte Tabellen sind schneller zum Schreiben oder Aufzeichnen von Daten.

3.3.2 Dimension Calculation Views

In diesem Abschnitt zeigen wir Ihnen, wie Sie einen Dimension Calculation View modellieren. Wie bereits erwähnt, werden Dimension Calculation Views zum Modellieren von Attributen verwendet. Der Fokus liegt auf diesen Attributen und nicht auf Kennzahlen. Kennzahlen werden in den Cube Calculation Views modelliert. Wir erstellen im Folgenden einen Dimension Calculation View, mit dem wir Kundendaten im Detail analysieren möchten.

Calculation View
anlegen

Für diese Aufgabe müssen Sie sich in der Perspektive SAP HANA Modeler befinden.

1. Legen Sie zunächst ein Paket unterhalb des Content-Ordners an und geben Sie dem Paket einen Namen, z. B. »ZSALES«, und vergeben Sie eine Beschreibung.
2. Öffnen Sie dann das Kontextmenü zu diesem Paket, und wählen Sie **New • Calculation View**.
3. Nennen Sie den View z. B. »CV_ZKUNDE«. CV_ steht hierbei für Calculation View. Wählen Sie den Typ **Graphical** und die Datenkategorie **DIMENSION** (siehe Abbildung 3.24). Klicken Sie dann auf **Finish**.

Abbildung 3.24 Dimension Calculation View anlegen

4. Der angelegte Dimension Calculation View wird Ihnen nun in der grafischen Modellierungsumgebung wie in Abbildung 3.25 angezeigt.

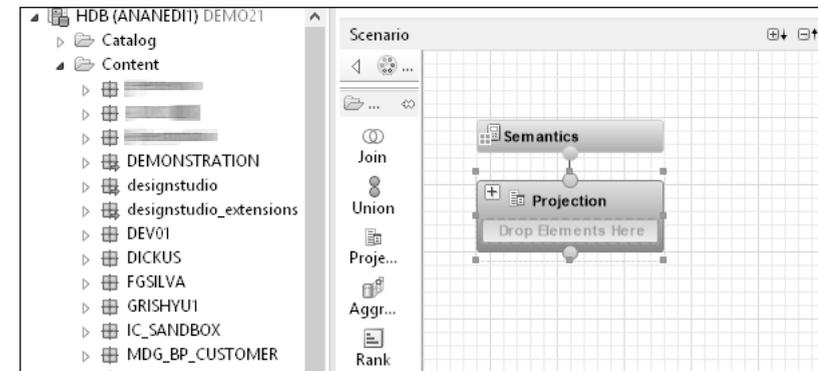


Abbildung 3.25 Grundgerüst des Calculation Views

Als Erstes modellieren Sie den Projektionsknoten (*Projection node*). Hier können Sie nur diejenigen Attribute der dem Calculation View zugrundeliegenden Tabelle oder Ansicht auswählen, die Sie projizieren, also in den Calculation View übernehmen möchten. Dazu müssen Sie zunächst die Quelle der Daten für die Modellierung angeben:

Projection Node

1. Klicken Sie auf das Pluszeichen neben dem Projektionsknoten, das Ihnen wie in Abbildung 3.26 gezeigt wird, wenn Sie den Mauszeiger über den Projektionsknoten bewegen.

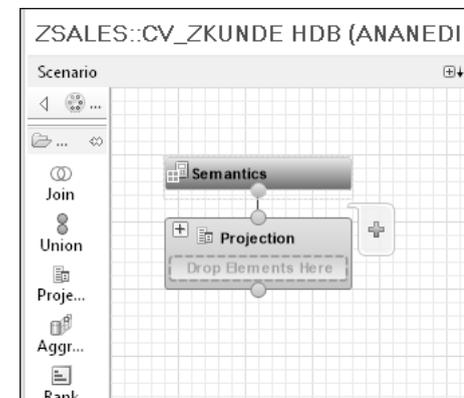


Abbildung 3.26 Klicken Sie auf Pluszeichen.

2. Es erscheint ein Fenster, in dem Sie nach dem Datenbankobjekt gefragt werden, das für die Modellierung verwendet werden soll. Für diese Übung verwenden wir die Tabelle KNA1 im Schema SAPHANADB, wie in Abbildung 3.27 gezeigt. Bestätigen Sie Ihre Eingabe.

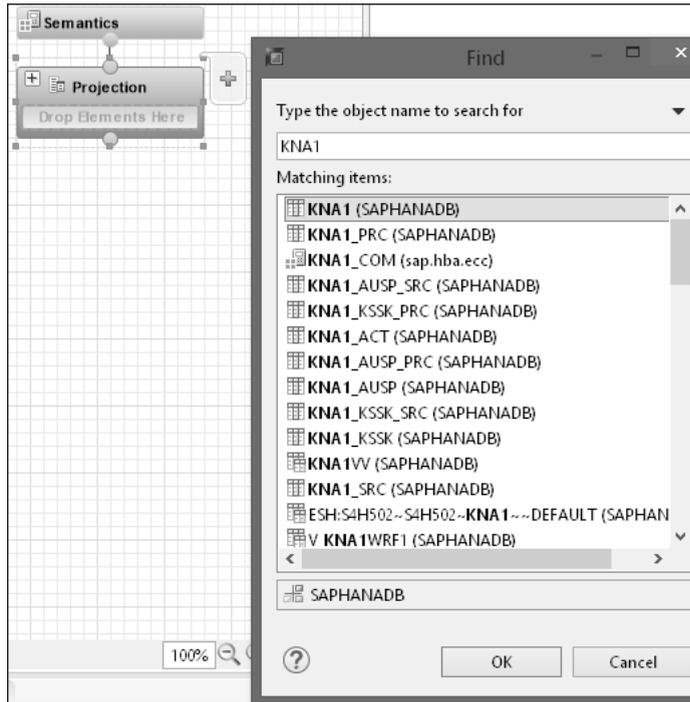


Abbildung 3.27 Datenquelle für die Modellierung hinzufügen

- In der Modellierungssicht wird nun die Tabelle im Projektionsknoten angezeigt. Neben der grafischen Modellierungssicht werden die Felder der ausgewählten Tabelle (die Attribute) in den **Details** angezeigt (siehe Abbildung 3.28).

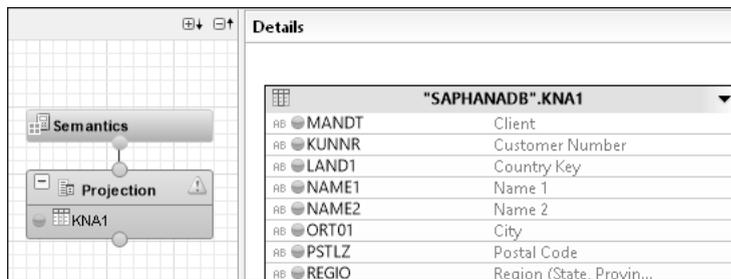


Abbildung 3.28 Informationen zur Datenquelle des Projektionsknotens

- Wählen Sie nun diejenigen Attribute aus, die Sie für Ihre Berichterstattung benötigen. Klicken Sie dazu auf ein Attribut, und wählen Sie **Add to Output** im Kontextmenü (siehe Abbildung 3.29).



Abbildung 3.29 Attribute auswählen

- Die ausgewählten Attribute werden nun neben den Details im Bereich **Output** angezeigt (siehe Abbildung 3.30).

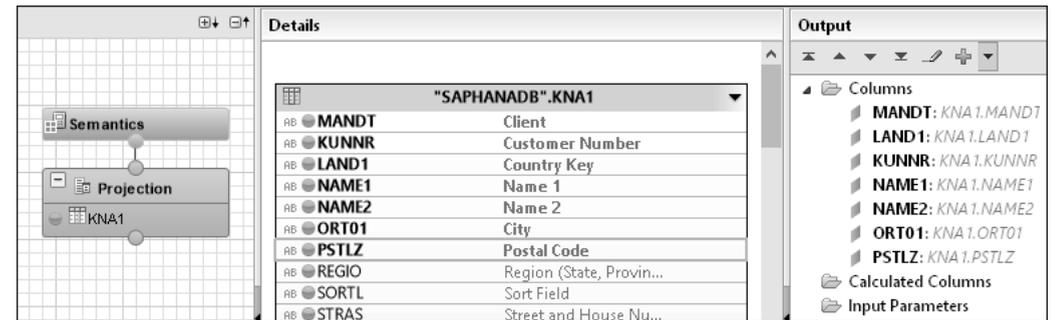


Abbildung 3.30 Ausgewählte Attribute

Über dem Projektionsknoten befindet sich der Semantiknoten (*Semantics Node*). Hier können Sie weitere Anpassungen vornehmen. Für unser Beispiel wählen Sie an dieser Stelle die Attributsschlüssel aus, indem Sie diese in der Sicht **Details** auf der Registerkarte **Columns** ankreuzen (siehe Abbildung 3.31). Definieren Sie die Attribute MANDT und KUNNR der Tabelle KNA1 als Schlüssel. Genau wie die Schlüssel der zugrundeliegenden Tabellen dienen die Schlüssel des Calculation Views zur eindeutigen Identifizierung der Datensätze.

Semantics Node

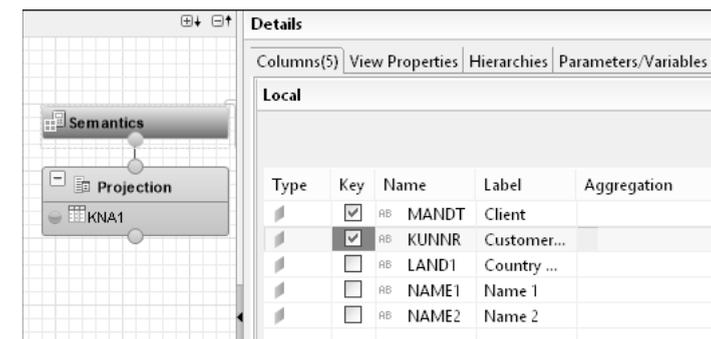


Abbildung 3.31 Attributsschlüssel im Semantiknoten anpassen

Unter **Name** sind die Attribute mit ihrem technischen Namen aufgelistet. Sie können sie auch umbenennen, wie wir in Abschnitt 4.1.6, »Technische Feldbezeichner ändern«, zeigen werden.

View-Eigenschaften

Auf der Registerkarte **View Properties** definieren Sie die allgemeinen Eigenschaften des Views. Ändern Sie den Eintrag im Feld **Default Client** von **Session Client** auf **Cross Client** (siehe Abbildung 3.32). Auf diese Weise sehen Sie für unsere Übung möglichst viele Daten.



Abbildung 3.32 Default Client ändern



Standard-Client

Es werden an dieser Stelle folgende Werte für den Standard-Client unterschieden:

- **Session Client**
Wenn Sie den Session Client als Standard-Client verwenden, filtert der Modellierer zur Laufzeit die Tabellendaten entsprechend dem Wert, den Sie für den Session Client im Benutzerprofil angegeben haben.
- **Cross Client**
Wenn Sie Cross Client als Standard-Client-Wert verwenden, filtert der Modellierer die Tabellendaten nicht nach einem Client, und Sie sehen Werte, die für alle Clients relevant sind.

Speichern und aktivieren

Abschließend aktivieren Sie den Calculation View, indem Sie in der Hauptfunktionsleiste auf das grüne Symbol mit dem Pfeil klicken und aus dem Auswahlménü die Option **Save and Activate** wählen (siehe Abbildung 3.33).

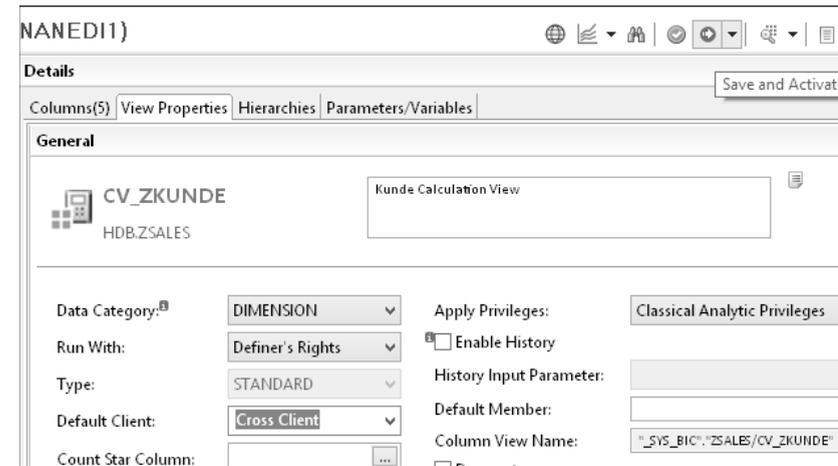


Abbildung 3.33 Dimension Calculation View speichern und aktivieren

Sobald Sie den Calculation View aktiviert haben, wird ein Column View erstellt. Den Namen dieses Column Views finden Sie auf der Registerkarte **View Properties** im Feld **Column View Name**. In unserem Beispiel lautet der Name `"_SYS_BIC"."ZSALES/CV_ZKUNDE"`.

Reportingwerkzeuge verwenden diesen Column View, um auf die Daten der SAP-HANA-Datenbank zuzugreifen. Wie im ersten Teil des Namens zu erkennen, wird der View im Schema `_SYS_BIC` erstellt. In Abbildung 3.34 sehen Sie ihn unten in dem Ordner **Column Views**.

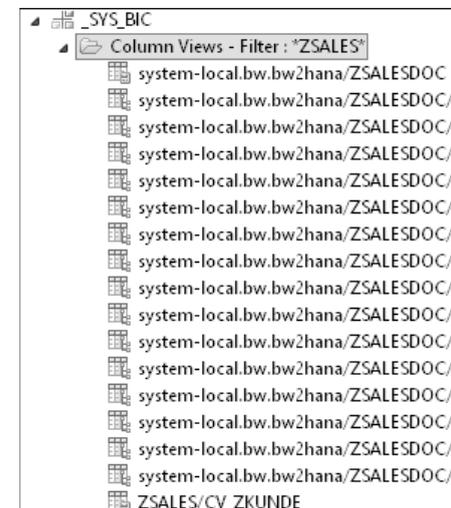


Abbildung 3.34 Der erstellte Column View im Schema `_SYS_BIC`

Daten anzeigen Sie haben nun zwei Möglichkeiten, um innerhalb der Perspektive SAP HANA Modeler die Daten des erstellten Modells anzuzeigen:

- Sie können die Datenvorschau nutzen. Klicken Sie dazu auf das Lupensymbol in der Hauptfunktionsleiste, und wählen Sie **Data Preview** (siehe Abbildung 3.35).



Abbildung 3.35 Data Preview (Datenvorschau)

Die Ergebnisse werden Ihnen daraufhin in einer Tabelle wie in Abbildung 3.36 angezeigt.

RB	MANDT	RB	KUNNR	RB	LAND1	RB	NAME1
100			0010100003		DE		Inlandskunde DE 3
100			0010100001		DE		Inlandskunde DE 1
100			0010100273		DE		Inlandskunde DE CPD
100			0010100002		DE		Inlandskunde DE 2
100			0010100004		DE		Inlandskunde DE 4
100			0010100006		DE		Inlandskunde DE 6
100			0010100005		DE		Inlandskunde DE 5 (CMS)
100			0010100100		DE		Inlandskunde DE 100 (SEPA)
100			0017100001		US		Domestic US Customer 1
100			0017100002		US		Domestic US Customer 2
100			0017100003		US		Domestic US Customer 3
100			0017100004		US		Domestic US Customer 4
100			0017100006		US		Domestic US Customer 6 (return)
100			0017100005		US		Domestic US Customer 5 (CMS)
100			0017100273		US		Domestic US CPD
100			0010100050		US		External Client 50 (US)
100			0010100051		FR		External Cleint 51

Abbildung 3.36 Datenvorschau des Column Views

- Alternativ können Sie eine SQL-Abfrage auf den Column View absetzen, um die Daten anzuzeigen. Dazu öffnen Sie die SQL-Konsole (SQL) über die Funktionsleiste der Registerkarte **Systems** und geben den folgenden Befehl ein:

```
Select * From "_SYS_BIC"."ZSALES/CV_ZKUNDE"
```

Die Ergebnisse werden Ihnen dann auf der Registerkarte **Result** angezeigt (siehe Abbildung 3.37).

	MANDT	KUNNR	LAND1	NAME1	NAME2
1	100	0010100003	DE	Inlandskunde DE 3	
2	100	0010100001	DE	Inlandskunde DE 1	
3	100	0010100273	DE	Inlandskunde DE...	
4	100	0010100002	DE	Inlandskunde DE 2	
5	100	0010100004	DE	Inlandskunde DE 4	
6	100	0010100006	DE	Inlandskunde DE 6	
7	100	0010100005	DE	Inlandskunde DE...	
8	100	0010100100	DE	Inlandskunde DE...	
9	100	0017100001	US	Domestic US Cu...	
10	100	0017100002	US	Domestic US Cu...	
11	100	0017100003	US	Domestic US Cu...	
12	100	0017100004	US	Domestic US Cu...	
13	100	0017100006	US	Domestic US Cu...	
14	100	0017100005	US	Domestic US Cu...	
15	100	0017100273	US	Domestic US CPD	
16	100	0010100050	US	External Client 5...	
17	100	0010100051	FR	External Cleint 51	

Abbildung 3.37 Ergebnisse der SQL-Abfrage

3.3.3 Dimension Calculation Views mit Text Join

Eine der Aufgaben bei der Modellierung besteht darin, den Endanwendern die Darstellung der gewünschten Daten in der Sprache zu ermöglichen, in der sie sich am System angemeldet haben. Wenn diese sich z. B. mit der Anmeldesprache Deutsch anmelden, wollen sie die Stammdaten auf Deutsch sehen. Melden sie sich auf Englisch an, möchten sie die Stammdateninformationen auf Englisch angezeigt bekommen.

Um diese Anforderung zu erfüllen, können Sie einen Dimension Calculation View und einen *Text Join* verwenden. Im Folgenden erklären wir das Vorgehen anhand der Tabellen MARA (allgemeine Materialdaten) und MAKT (Materialbeschreibungen). Die Tabelle MARA enthält alle Informationen zu einem Material, und die Tabelle MAKT enthält die Beschreibungen in verschiedenen Sprachen. Sie hat ein Schlüsselfeld namens SPRAS (Sprachenschlüssel).

Text Join

Dies ist in Abbildung 3.38 zu erkennen. Das Feld SPRAS wird verwendet, um die Materialbeschreibung abhängig von der SAP-Anmeldesprache anzuzeigen.

Field	Key	Ini...	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
MATNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MATNR	CHAR	40	0	Material Number
SPRAS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SPRAS	LANG	1	0	Language Key
MAKTX	<input type="checkbox"/>	<input type="checkbox"/>	MAKTX	CHAR	40	0	Material description
MAKTG	<input type="checkbox"/>	<input type="checkbox"/>	MAKTG	CHAR	40	0	Material Description

Abbildung 3.38 Feld SPRAS (Sprachenschlüssel) in der Tabelle MAKT

Wenn Sie sich den Inhalt der Tabelle MAKT anschauen, sehen Sie, dass es für das gleiche Material Beschreibungen in unterschiedlichen Sprachen gibt (siehe Abbildung 3.39).

Cl.	Material	Language	Material description
501	FERT1000	B	FIN126,MTS-DI,PD,מספר סידורי, מוספר סידורי
501	FERT1000	C	FIN126,MTS-DI,PD,řarže-FIFO,sér. číslo
501	FERT1000	D	FERT, Auftragsn, PD, FIFO Charge, WM
501	FERT1000	E	FIN126,MTS-DI,PD,Batch-Fifo,SerialNo
501	FERT1000	F	FIN126,MTS-DI,PD,FIFOpárLot,Nřsérie
501	FERT1000	G	FIN126,MTS-DI,PD,Παρτίδα-Fifo,Σειριακό

Abbildung 3.39 Materialbeschreibungen in unterschiedlichen Sprachen

Wir werden nun die Tabellen MARA und MAKT in einem Dimension Calculation View verknüpfen. Dadurch sehen die Endanwender dann Materialinformationen in ihrer Anmeldesprache.

Anmeldesprache ändern

Wir werden den Information View testen, indem wir die Anmeldesprache ändern. Daher zeigen wir Ihnen kurz, wie man die Anmeldesprache im SAP-HANA-System ändert. Markieren Sie dazu auf der Registerkarte **Systems** in SAP HANA Studio Ihr SAP-HANA-System, und öffnen Sie das Kontextmenü.

Wählen Sie den Eintrag **Properties** (siehe Abbildung 3.40), um die Eigenschaften des Systems anzuzeigen.

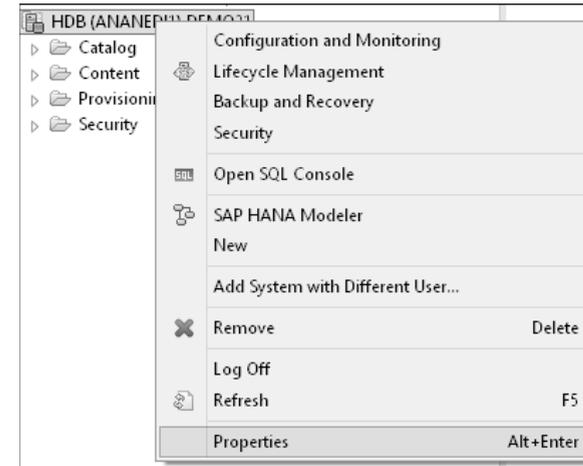


Abbildung 3.40 Eigenschaften des SAP-HANA-Systems in SAP HANA Studio aufrufen

In unserem Beispiel ist unter **Database User Logon • Additional Properties** im Feld **Locale** die Anmeldesprache **German** eingestellt (siehe Abbildung 3.41).

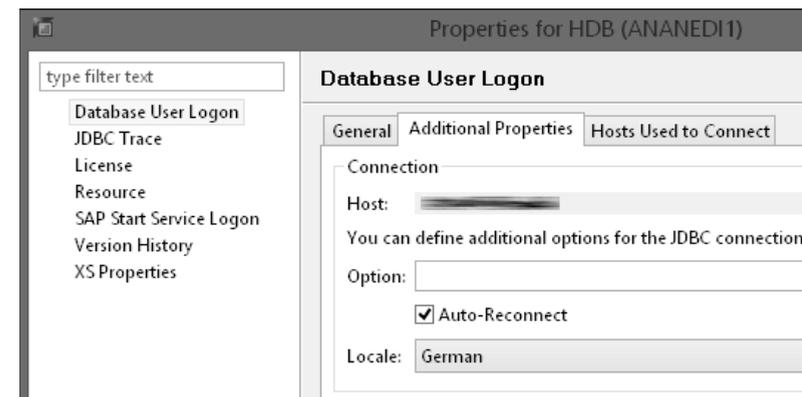


Abbildung 3.41 Der Benutzer ist mit der Sprache Deutsch angemeldet.

Erstellen Sie nun erneut einen Dimension Calculation View in Ihrem Paket, wie im vorangehenden Abschnitt gezeigt. Nennen Sie den View z. B. »ZMAT_TEXT«, und vergeben Sie die Beschreibung »Material Text«.

Calculation View erstellen

Für unsere Zwecke benötigen wir nicht alle Attribute in den Tabellen MARA und MAKT. Deshalb werden wir über die Projektionsknoten nur die Attribute auswählen, an denen wir interessiert sind.

Attribute einschränken

1. Fügen Sie dazu zunächst zwei zusätzliche Projektionsknoten hinzu. Dazu wählen Sie in der Werkzeugleiste (unter **Scenario**) auf der linken Seite der grafischen Modellierungsumgebung den Projektionsknoten aus und ziehen ihn auf die Arbeitsfläche (siehe Abbildung 3.42). Wiederholen Sie das Vorgehen für den zweiten zusätzlichen Projektionsknoten. Das System vergibt automatisch die Namen **Projection_1** und **Projection_2** für die Knoten.

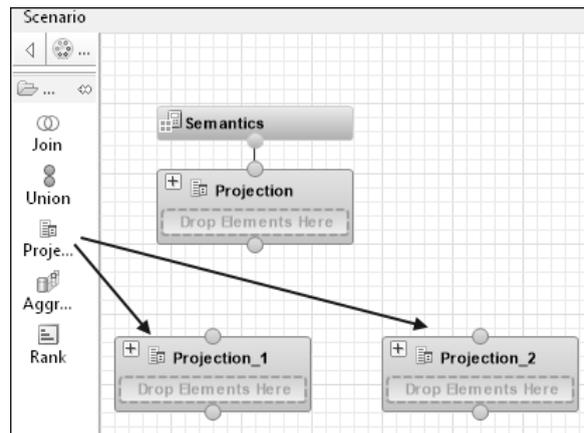


Abbildung 3.42 Zwei zusätzliche Projektionsknoten hinzufügen

2. Dem Projektionsknoten **Projection_1** fügen Sie nun die Tabelle **MARA** und dem Projektionsknoten **Projection_2** die Tabelle **MAKT** als Datenquelle hinzu, wie wir das im vorangehenden Beispiel anhand der Tabelle **KNA1** gezeigt haben. Das Ergebnis sehen Sie in Abbildung 3.43.

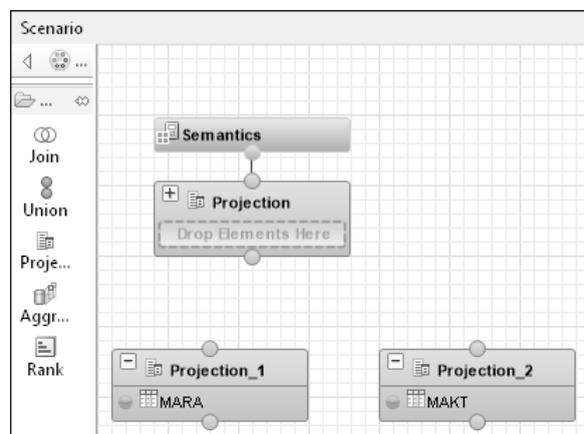


Abbildung 3.43 Tabelle MARA und MAKT als Datenquellen

3. Klicken Sie nun auf den Projektionsknoten **Projection_2**, und wählen Sie die Attribute **MANDT**, **MATNR**, **SPRAS**, **MAKTX** und **MAKTG** aus, also alle Felder der Tabelle **MAKT** (siehe Abbildung 3.44).

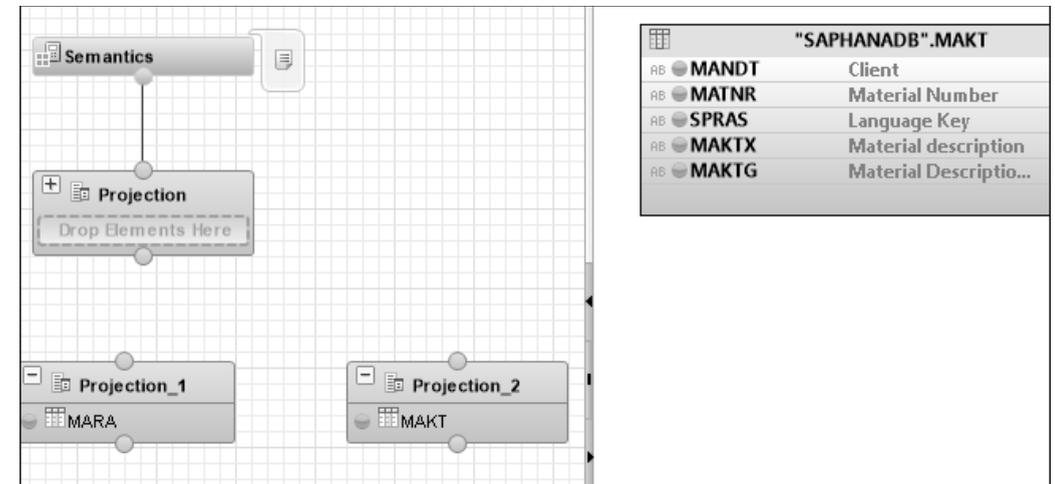


Abbildung 3.44 Attribute der Tabelle MAKT auswählen

4. Aus den Feldern der Tabelle **MARA** wählen Sie **MANDT**, **MATNR** und **MTART** aus (siehe Abbildung 3.45).

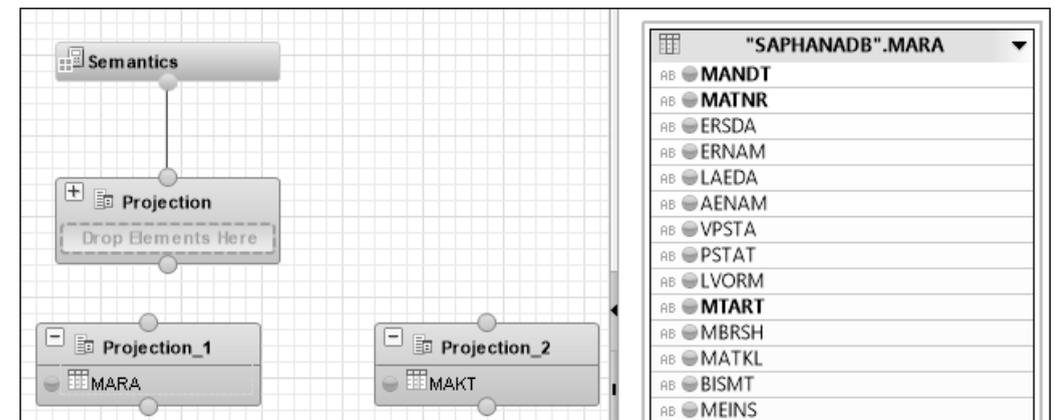


Abbildung 3.45 Attribute der Tabelle MARA auswählen

5. Sie können die Knoten **Projection_1** und **Projection_2** nun noch umbenennen. Dazu öffnen Sie das Kontextmenü zu einem der Knoten und wählen die Funktion **Rename**. Wir haben die Knoten in **Material** und **MAT_TEXT** umbenannt (siehe Abbildung 3.46).

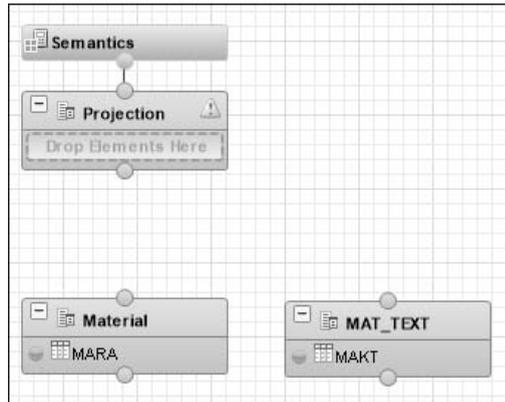


Abbildung 3.46 Projektionsknoten umbenennen

Join Node Jetzt benötigen wir einen Join-Knoten (*Join Node*), um die beiden Tabellen zu verknüpfen. Wählen Sie dazu in der Werkzeugleiste links die Funktion **Join** aus, und ziehen Sie den Knoten auf die Modellierfläche. Sie sehen den neuen Knoten in Abbildung 3.47.

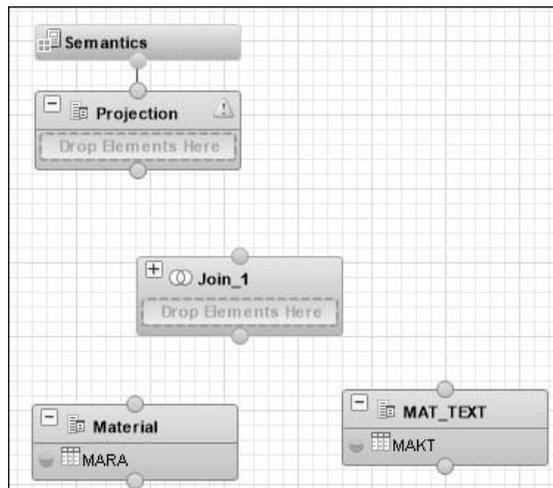


Abbildung 3.47 Join-Knoten hinzufügen

Verknüpfung herstellen Nun ziehen Sie jeweils eine Verbindungslinie von den Projektionsknoten zu dem Join-Knoten (siehe Abbildung 3.48). Dazu klicken Sie auf den Kreis oben am Projektionsknoten und ziehen die Verbindungslinie bei gedrückter Maustaste.

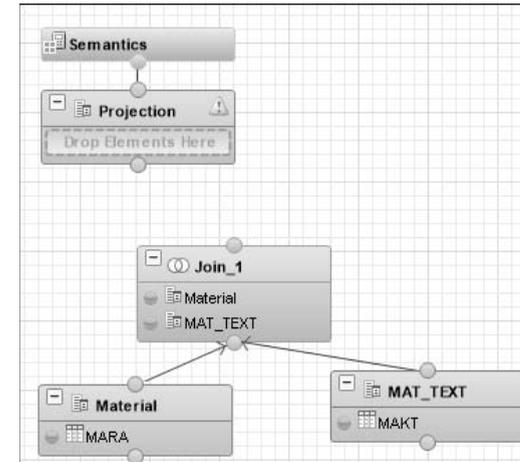


Abbildung 3.48 Projektionsknoten mit dem Join-Knoten verbinden

Die Namen der beiden Projektionsknoten werden nun auf dem Join-Knoten angezeigt. In dem Join-Knoten werden nun die Tabellen miteinander verknüpft. Markieren Sie dazu den Join-Knoten, und ziehen Sie im rechten Fenster **Details** mit gedrückter Maustaste jeweils eine Linie von den Attributen **MANDT** und **MATNR** des Knotens von **Material** zu den gleichnamigen Attributen des Knotens **MAT_TEXT**, wie in Abbildung 3.49 gezeigt.

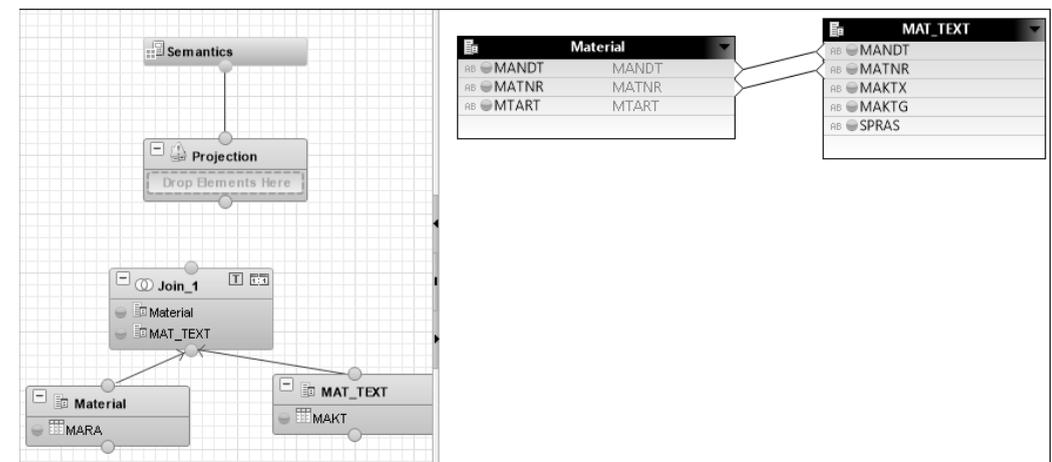


Abbildung 3.49 Attribute der Tabellen verbinden

Wenn Sie auf die Linie, die die **MANDT**-Felder verbindet, klicken, öffnet sich das Fenster **Edit Join** (siehe Abbildung 3.50). Ändern Sie hier den Join-Typ von der Standardeinstellung **Inner Join** zu **Text Join**. Wählen Sie im Feld **Language Column** das Feld **SPRAS**.

Join-Typ ändern

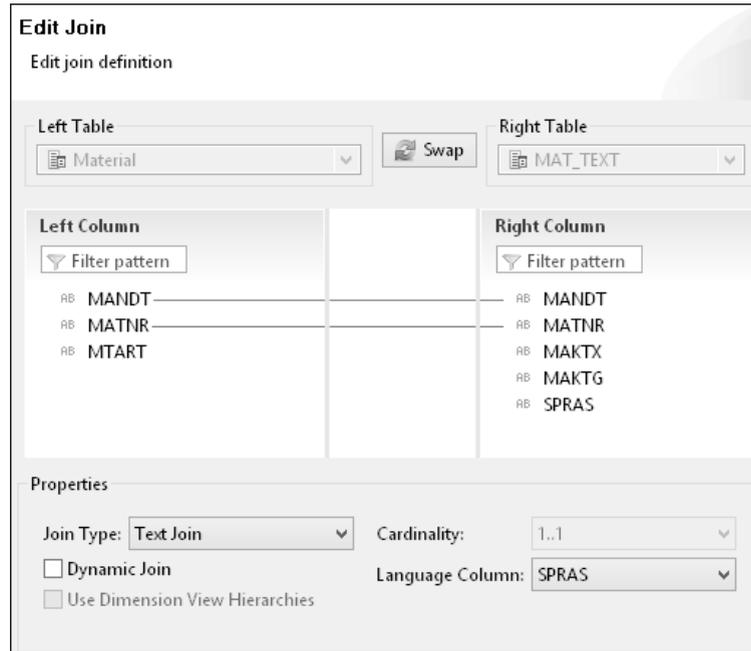


Abbildung 3.50 Join-Typ ändern

Stellen Sie sicher, dass die Attribute, die Sie in der Ausgabe sehen wollen, auch auf der Join-Ebene ausgewählt sind. Klicken Sie auf die in Abbildung 3.51 markierten Attribute.

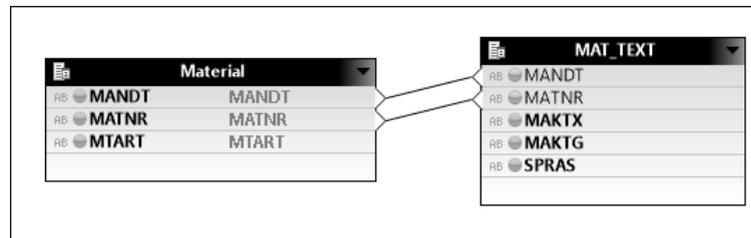


Abbildung 3.51 Attribute auswählen

Verbinden Sie nun in der grafischen Modellierungssicht den Join-Knoten mit dem oberen Projektionsknoten (siehe Abbildung 3.52). Markieren Sie dann den Join-Knoten, und wählen Sie im rechten Fenster alle Attribute aus (siehe Abbildung 3.53).

Eigenschaften des Calculation Views anpassen

Nun können Sie noch die Eigenschaften des Calculation Views anpassen. Klicken Sie dazu auf den semantischen Knoten. Sie müssen wie üblich die Schlüsselattribute auswählen. Unter **View Properties** ändern Sie wieder die

Einstellung für den **Default Client** auf **Cross Client**. Zuletzt aktivieren Sie den Dimension Calculation View.

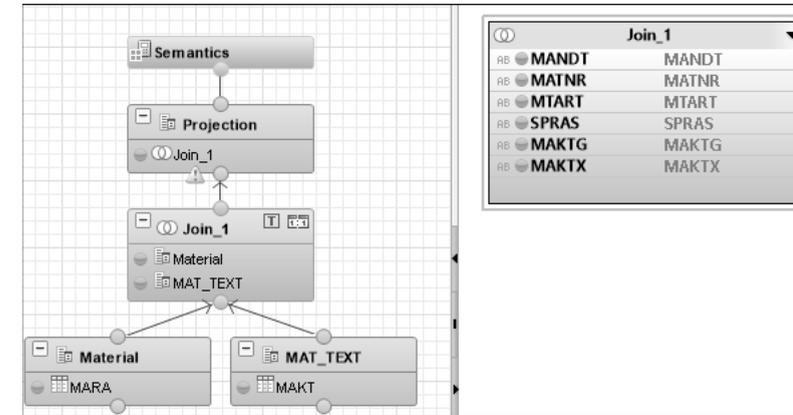


Abbildung 3.52 Join-Knoten mit dem ersten Projektionsknoten verbinden

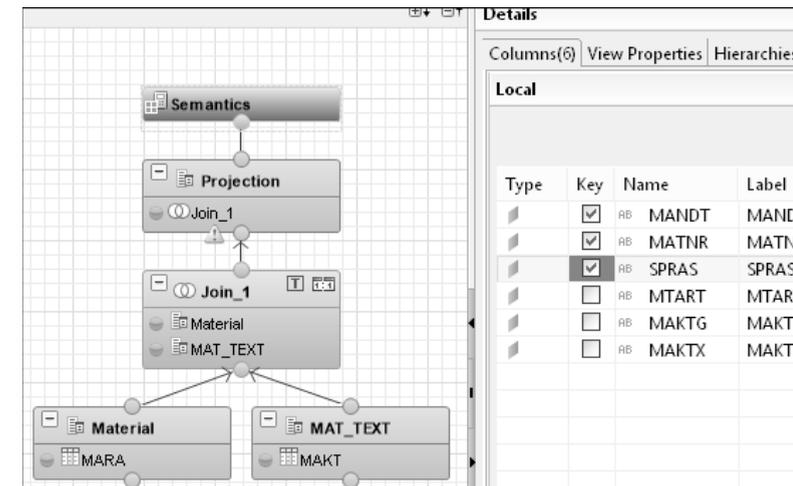


Abbildung 3.53 Schlüsselattribut im semantischen Knoten wählen

Wenn Sie nun die Datenvorschau für den View aufrufen, können Sie sehen, dass die Beschreibungen Felder in der Spalte MAKTG auf Deutsch angezeigt werden. Auch der Wert in der Spalte SPRAS für die Anmeldesprache ist »D« für Deutsch (siehe Abbildung 3.54).

Datenvorschau

Ändern Sie nun die Anmeldesprache auf Englisch, wie zu Beginn dieses Abschnitts gezeigt. Jetzt werden die Beschreibungen auf Englisch angezeigt, und der Wert in der Spalte SPRAS lautet »E« (siehe Abbildung 3.55).

RB	MANDT	RB	MATNR	RB	SPRAS	RB	MTART	RB	MAKTG
100			TG10		D		HAWA		HANDELSWARE 10, PD, STRECKE
100			TG11		D		HAWA		HANDELSWARE 11, PD, NORMALER HANDEL
100			TG12		D		HAWA		HAWA 12, BESTELLPUNKT, NORMALER HANDEL
100			TG13		D		HAWA		HANDELSWARE 13, BESTELLPUNKT, STRECKE
100			TG14		D		HAWA		HANDELSWARE 14, PD, ZUKAUF, H14
100			SG25		D		HALB		SEMI25, PD, FREMDBESCHAFFUNG
100			TG20		D		HAWA		HANDELSWARE 20, BESTELLPUNKT, SERIALNR.
100			TG22		D		HAWA		HAWA 22, BESTELLPKT, VERFALLDATUM CHARGE
100			TG0001		D		MAT		HANDELSWARE FÜR VERBRAUCH
100			TG0011		D		MAT		HANDELSWARE 0011, PD, REGULÄRE BESCHAFF.
100			TG0012		D		MAT		HANDWARE 0012, MELDEBEST., REG. BESCH.
100			SG325A		D		HALB		SEMI325A, PLM, PD, FREMDBESCHAFFUNG
100			FG226		D		FERT		FIN226, MTO, PD, FIFO CHARGE, SERIALNR.
100			SG325B		D		HALB		SEMI325B, ECM, PD, FREMDBESCHAFFUNG
100			RP001		D		LEIH		LEERGUT, ND

Abbildung 3.54 Materialtexte in deutscher Sprache

RB	MANDT	RB	MATNR	RB	SPRAS	RB	MTART	RB	MAKTG	RB	MAKTX
100			TG10		E		HAWA		TRAD.GOOD 10,PD,THIRD PARTY		Trad.Good 10...
100			TG11		E		HAWA		TRAD.GOOD 11,PD,REG.TRADING		Trad.Good 11...
100			TG12		E		HAWA		TRAD.GOOD 12,REORDER POINT,REG.TRA...		Trad.Good 12...
100			TG13		E		HAWA		TRAD.GOOD 13,REORDER POINT,THRD P...		Trad.Good 13...
100			TG14		E		HAWA		TRAD.GOOD 14,PD,BOUGHT-IN,H14		Trad.Good 14...
100			SG25		E		HALB		SEMI25,PD,EXTERNALPROCUREMENT		SEMI25,PD,Ex...
100			TG20		E		HAWA		TRAD.GOOD 20,REORDER POINT,SERIALNO		Trad.Good 20...
100			TG22		E		HAWA		TRAD.GOOD 22,REORDER POINT,BATCH-...		Trad.Good 22...
100			TG0001		E		MAT		TRADING GOOD FOR CONSUMPTION		Trading Goo...
100			TG0011		E		MAT		TRADING GOOD 0011,PD,REGULAR PROC.		Trading Goo...
100			TG0012		E		MAT		TRAD. GOOD 0012,REORD. PT.,REGULAR P...		Trad. Good 0...
100			SG325A		E		HALB		SEMI325A,PLM,PD,EXTERNALPROCUREM...		SEMI325A,PL...
100			FG226		E		FERT		FIN226,MTO,PD,BATCH-FIFO,SERIALNO		FIN226,MTO,...
100			SG325B		E		HALB		SEMI325B,ECM,PD,EXTERNALPROCUREM...		SEMI325B,EC...
100			RP001		E		LEIH		EMPTIES,ND		Empties,ND

Abbildung 3.55 Materialtexte in englischer Sprache

3.4 Fazit

In diesem Kapitel haben wir Sie in die native Modellierung von Informationsmodellen in SAP HANA eingeführt. Sie haben die Modellierungsumgebung in SAP HANA Studio kennengelernt, die Perspektive **SAP HANA Modeler**. Anschließend haben wir uns mit der Modellierung von Dimension Calculation Views beschäftigt. Im folgenden Kapitel gehen wir auf die Modellierung weiterer Calculation-View-Typen ein. Unter anderem zeigen wir Ihnen die Modellierung von Cube Calculation Views, zeitbasierten Calculation Views und verschiedenen fortgeschrittenen Funktionen wie Währungsumrechnung, Referential und Star Joins, Filter und Ranglisten.