

## Kapitel 2

# Skalierbare Architektur für Business Intelligence

*»With HANA, we are back to the seventies with regards to the true real-time enterprise. Much of the reporting and analytics functionality can now come back and run directly on the normalized transaction data.«*  
– Hasso Plattner und Bernd Leukert, »The In-Memory Revolution«, Springer 2015

In diesem Kapitel beschäftige ich mich mit der Architektur von SAP BW/4HANA. Im Zentrum dieser Architektur steht die Skalierbarkeit. Die Namensgebung, »BW/4HANA« anstelle von »BW auf HANA«, gibt zu verstehen, dass die SAP-HANA-Datenbank mit der neuen Data-Warehousing-Lösung nicht mehr als *Blackbox* angesehen wird, in der nur der Systembenutzer des SAP-BW/4HANA-Servers Befehle absetzt. Vielmehr werden die Modellierungsmöglichkeiten der SAP-HANA-Datenbank genutzt, um ein Maximum an Performance und analytischer Leistung zu erbringen.

SAP BW/4HANA wird nicht als Teil des SAP-NetWeaver-Stack ausgeliefert. Die zugrundeliegenden Plattformen sind SAP HANA sowie ein im Vergleich zu dem Applikationsserver für SAP BW verschlankter ABAP-Applikationsserver. Die Modellierungsoptionen des SAP-BW/4HANA-Servers und der SAP-HANA-Datenbank ergänzen sich, so dass eine leistungsfähige Lösung für das Enterprise Data Warehousing entsteht. Bis heute betreiben insbesondere größere Einzelhändler neben SAP BW ein zusätzliches, SQL-basiertes Enterprise Data Warehouse. Es besteht nun die Möglichkeit, dass diese Koexistenz zukünftig durch SAP BW/4HANA abgelöst wird. Denn SAP BW/4HANA unterstützt neben dem ABAP-orientierten Datenmanagement auch klassisches, d. h. SQL-basiertes Data Warehousing.

In diesem Kapitel vermittele ich Ihnen zunächst einen Überblick über die Architektur von SAP BW/4HANA und den damit verbundenen BI-Clients (Business Intelligence). Anschließend stelle ich Ihnen die Modellierungsobjekte von SAP BW/4HANA vor, und zeige Ihnen, wie Sie diese Objekte im SAP HANA Studio anlegen. Abschließend finden Sie einige Informationen über die Werkzeuge der SAP HANA Data Warehousing Foundation.

## 2.1 Die Architektur von SAP BW/4HANA im Überblick

Bereits seit den ersten Data Warehouses beschäftigt man sich mit dem Problem der Skalierbarkeit von Architekturen, um die Anforderungen im Bereich BI erfüllen zu können. Für SAP BW wurde daher die Layered Scalable Architecture (LSA) als Referenzarchitektur formuliert. Diese Referenzarchitektur stellt sicher, dass die Anforderungen an das Datenmanagement sowie der Datenanalyse erfüllt werden:

- Beim *Datenmanagement* geht es um die effiziente Verwaltung der Daten, also deren Beschaffung, Strukturierung und Aktualisierung.
- Die *Datenanalyse* erfordert eine effiziente Auswertbarkeit der Daten, also die Möglichkeit, relevante Daten zu selektieren und analytisch aufzubereiten.

In beiden Fällen spielt der Zeitfaktor eine entscheidende Rolle. Neue Daten bieten den größten Mehrwert für die Fachbereiche. Stehen die Daten nicht rechtzeitig zur Verfügung, geraten Geschäftsprozesse ins Stocken. Genau hier setzt SAP BW/4HANA an: Mithilfe logischer Modellierungskonzepte sollen Daten möglichst in Echtzeit und ohne Ladeprozesse direkt ausgewertet werden können. Die Verschmelzung von Online Analytical Processing (OLAP) und Online Transaction Processing (OLTP) war folglich ein erklärtes Ziel bei der Entwicklung von SAP HANA. Abbildung 2.1 zeigt die Kernkomponenten von SAP BW/4HANA zusammen mit den Schnittstellen zu den Quellsystemen und den BI-Frontend-Werkzeugen. Zu den Kernkomponenten gehören die BW-Modellierungswerkzeuge (*BW Modeling Tools*), die die *Data Warehousing Workbench* früherer SAP-BW-Releases nach und nach ablösen, sowie der SAP-BW/4HANA-Server, der alle ABAP-basierten Objekte der SAP-BW/4HANA-Architektur verwaltet.

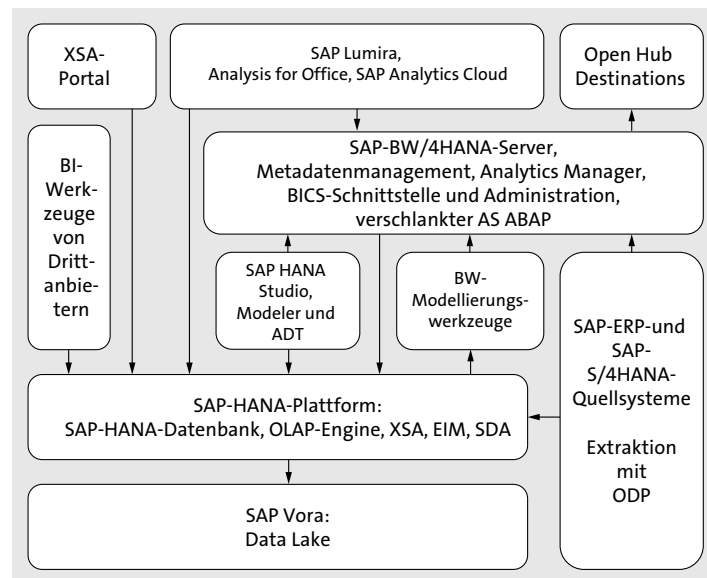


Abbildung 2.1 Architektur von SAP BW/4HANA

SAP BW/4HANA macht klassisches Data Warehousing nicht überflüssig. Die Hauptaufgabe des Data Warehousing, nämlich die Integration von Daten aus verschiedenen Quellen, bleibt weiterhin bestehen. Bevor eine Entität wie ein Kunde oder Artikel erstellt werden kann, müssen deren Bestandteile zunächst auf Datenebene technisch verfügbar sein. Bei unterschiedlichen Quellen müssen also deren Schlüssel, Texte, Stammdaten und Hierarchien zusammengeführt werden. In früheren SAP-BW-Releases war ABAP die Sprache dieser Transformationen. Die Sprache der SAP-HANA-Datenbank hingegen ist *SQLScript*.

Es gibt bereits die Möglichkeit, SAP-BW/4HANA-Transformationen in der SAP-HANA-Datenbank auszuführen. Voraussetzung ist die Formulierung der Transformation als direkte Abbildung oder Formel. Die Semantik dieser Transformationen lässt sich mithilfe eines Compilers in sogenannte *Kalkulationsszenarien* übersetzen. Schwerlich möglich ist hingegen die Übersetzung komplexer ABAP-Transformationen. Falls Formeln Ihren Sachverhalt nicht abbilden können, ist es sinnvoll, Transformationen für SAP BW/4HANA in *SQLScript* zu formulieren. In Abschnitt 4.5.3, »Transformationen mit ABAP Managed Database Procedures entwickeln«, wird diese Art der Transformation beschrieben.

Grundsätzlich ging das Data-Warehousing-Konzept immer von vorstrukturierten Daten aus. Schließlich stammt das Konzept aus einer Zeit vor dem explosionsartigen Wachstum des Internets. Allerdings hat sich die Welt der Daten in den vergangenen knapp 40 Jahren erheblich verändert, so dass neue Konzepte erforderlich sind. Das Data-Lake-Konzept, das in Kapitel 7 ausführlich beschrieben wird, scheint diesen neuen Anforderungen gewachsen zu sein. Doch auch wenn es so möglich ist, die Datenvielfalt zu bewältigen, fehlt die für die Fachbereiche notwendige Strukturierung, denn ohne Struktur kann keine effiziente Auswertung der Daten erfolgen. Hier wird dem Data-Warehousing-Konzept, das auf eine jahrzehntelange Erfahrung mit vorstrukturierten Daten zurückblicken kann, neues Leben eingehaucht.

Insgesamt lassen sich drei Schichten der SAP-BW/4HANA-Architektur identifizieren, die in Abbildung 2.2 dargestellt sind:

- semantische Schicht
- Integrationsschicht
- Vorratsschicht

Die skalierbare Architektur beinhaltet die LSA früherer SAP-BW-Releases, umfasst jedoch alle Ebenen bis hinunter zum Data Lake. Details zur erweiterten LSA erfahren Sie in Abschnitt 4.1.1, »Model-View-Controller und Layered Scalable Architecture (LSA++)«.

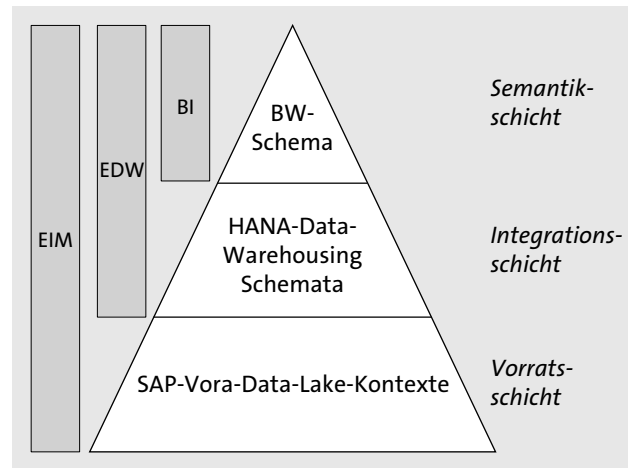


Abbildung 2.2 Skalierbare Architektur für Business Intelligence

Zunächst möchte ich kurz reflektieren, weshalb ein BW-veraltetes Schema im Data Warehousing Vorteile mit sich bringt. Grundsätzlich ist Data Warehousing kein Produkt, sondern ein permanenter Prozess der Anpassung des Datenbestands an die sich fortlaufend ändernden Analyseprozesse.

Abhängig von dem zu analysierenden Datenbestand schwankt die Änderungsgeschwindigkeit dieser Analysen. Die geringste Änderungshäufigkeit erfahren sicherlich Analysen aus dem Bereich des Finanzwesens (FI), da der Gesetzgeber hier Vorgaben hinsichtlich der Buchungsperioden, Kontenpläne etc. macht. Im Bereich des Personalwesens (HR) wird es dynamischer, da dieser Bereich auch aktuellen Trends im Personalmanagement gerecht werden muss. In der Logistik kann man bereits in Ansätzen von agilen Geschäftsprozessen ausgehen, da veränderte Marktchancen oder politische Ereignisse unter Umständen auch kurzfristige Anpassungen erfordern. Der Bereich Marketing muss gänzlich agil geführt werden, da Aktionen sehr kurzfristig auf Wetterperioden, Aktionen der Wettbewerber oder auch einmal auf ein virales YouTube-Video reagieren müssen. Hier würde ein starres, vorgegebenes Schema die Geschäftstätigkeit eher behindern als fördern.

Als extreme Ausprägung der agilen Datenanalyse hat sich die Datenwissenschaft (*Data Science*) herausgebildet, die gänzlich ohne vorgefertigte Strukturen auskommt und Analysen auf Basis der Rohdaten ausführen muss. Abbildung 2.3 zeigt die gegenläufigen Abhängigkeiten (Trades-Offs) zwischen Effizienz und Agilität.

Daraus wird deutlich, dass ein BW-veraltetes Schema nur in Bereichen mit geringer Agilität sinnvoll ist. Gleichzeitig bietet ein solches Schema eine Schnittstelle zur strukturierten Übergabe von Ergebnissen an agile Abteilungen, so dass sie einen Beitrag zur Entscheidungsunterstützung liefern können.

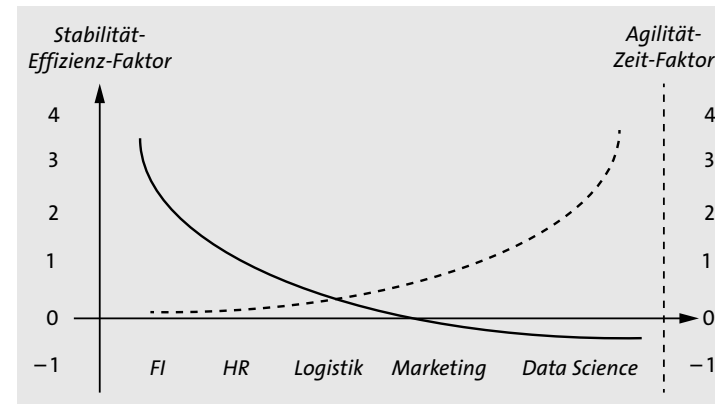


Abbildung 2.3 Skalierbare Architektur – Effizienz vs. Agilität

## 2.2 Die BI-Clients für SAP BW/4HANA

Im Zentrum der Analysewerkzeuge für SAP BW/4HANA stehen die BI-Client-Anwendungen von SAP BusinessObjects. Das Peacock-Diagramm in Abbildung 2.4 zeigt die einsetzbaren BI-Clients und Schnittstellen im Überblick. Die optionale *SAP BusinessObjects BI Platform* wird dabei nicht berücksichtigt.

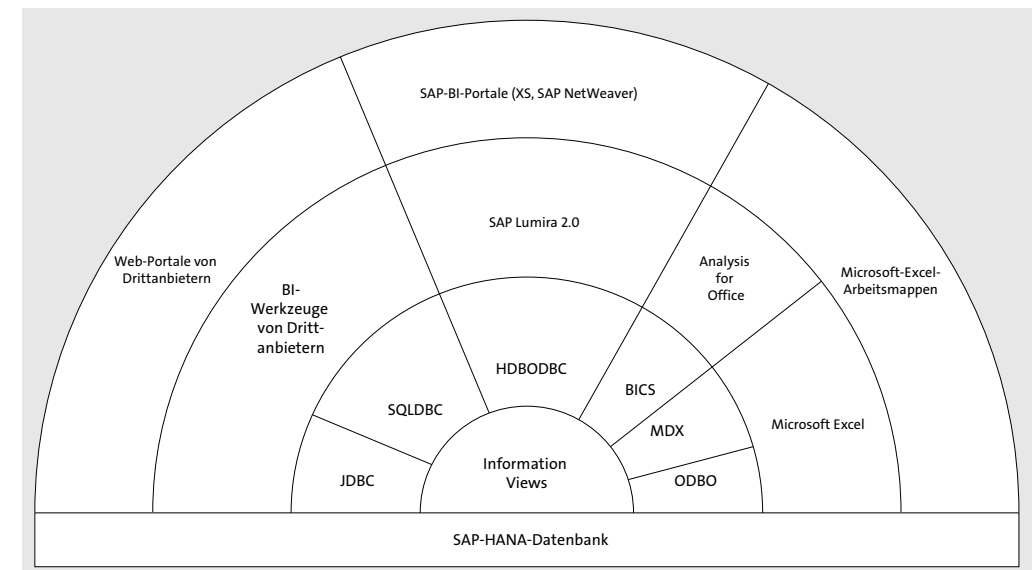


Abbildung 2.4 SAP-BW/4HANA-Schnittstellen und Client-Anwendungen

Natürlich gibt es auch BI-Client-Anwendungen von Drittanbietern, die *Multidimensional Expressions* (MDX) über eine ODBO-Verbindung (*OLE DB for OLAP*) schreiben

und lesen können. In der Praxis nutzen diese Werkzeuge jedoch zumeist die xDBC-Schnittstellen (also JDBC oder ODBC) und generieren SQL-Code direkt für die SAP-HANA-Datenbank.

In Abschnitt 3.3, »Installation der Client-Werkzeuge für SAP HANA«, zeige ich Ihnen die Konfiguration des HDBODBC-Treibers, der zusammen mit den SAP-HANA-Client-Tools installiert wird. Der HDBODBC-Treiber basiert auf dem performanteren SQLDBC-Treiber, dessen Protokoll von SAP ursprünglich für die Datenbank SAP MaxDB (<http://maxdb.sap.com/>) entwickelt wurde. Er ist durch seine C++-Basis insbesondere für die native Kommunikation von Anwendungen mit der SAP-HANA-Datenbank konzipiert. Da ODBO und MDX Microsoft-Standards sind, eignen sie sich besonders gut zur Kommunikation mit Microsoft Excel.

Die *BI Consumer Services* (BICS) werden vom SAP-BW/4HANA-Server weiterhin unterstützt. So können bestehende BW Queries, die in früheren SAP-BW-Releases erstellt wurden, weiterhin genutzt und erweitert werden. BICS können von SAP BusinessObjects Analysis for Microsoft Office konsumiert werden. In Abschnitt 2.4.16, »Arbeitsmappe in SAP BusinessObjects Analysis for Microsoft Office anlegen«, zeige ich Ihnen, wie Sie im SAP HANA Studio eine Query mit den BW-Modellierungswerkzeugen anlegen und mit Analysis for Office in eine Arbeitsmappe einbinden. Weitere BI-Analyse-Werkzeuge stelle ich Ihnen in Kapitel 5, »Frontend-Tools für Business Intelligence«, vor.

Im Bereich der webbasierten BI-Portale gibt es zahlreiche neue Optionen, da insbesondere mit dem Anwendungsserver SAP HANA Extended Application Services, Advanced Model (XSA) alle Möglichkeiten der Gestaltung moderner Websites unterstützt werden. Detailliertere Informationen dazu finden Sie ebenfalls in Kapitel 5.

## 2.3 Aufgaben von SAP BW/4HANA

Die Aufgaben von SAP BW/4HANA können drei Gruppen zugeordnet werden:

### ■ Semantische Verbindung mit den SAP-ERP-Systemen

SAP ERP zeichnet sich durch eine Vielzahl semantischer Abhängigkeiten aus, die für das Reporting berücksichtigt werden müssen. Bekannte Beispiele sind die Währungsumrechnung oder die Konsolidierung von Umsatzzahlen im Konzern (Binnenumsatzelemierung).

Der *BI Content* in SAP BW/4HANA bildet die Objekte in den zugrundeliegenden SAP-ERP-Systemen ab. Ohne dessen Extraktoren müssten Sie deren ABAP-Code bzw. den SQLScript-Code studieren und bei jedem neuen Release der SAP Business Suite oder von SAP S/4HANA Ihr gesamtes Reporting überprüfen und einem Regressionstest unterziehen. Natürlich gibt es bei jedem Releasewechsel eine Testphase; der Aufwand wäre ohne den BI Content jedoch ungleich größer.

### ■ Integration der Geschäftsprozesse im Data Warehouse

SAP-ERP-Anwendungen sind häufig voneinander isoliert. Dies liegt entweder in der Anwendung selbst begründet oder wird explizit herbeigeführt, da nicht alle vernetzten Komponenten gleichzeitig weiterentwickelt werden können. Die Aufgabe von SAP BW/4HANA ist es dann, die Daten dieser verteilten Geschäftsprozesse zusammenzuführen, um ein einheitliches Bild der Geschäfte für das Management zu zeichnen.

### ■ Weiterentwicklung der Geschäftsprozesse

Aktuell sind noch sehr wenige Geschäftsprozesse in SAP ERP digitalisiert. Üblicherweise bedienen sich Entscheider des operativen oder strategischen Reporting, interpretieren die Zahlen und planen dann die nächsten Schritte in SAP ERP manuell oder teilautomatisiert. Für dieses *Closed-Loop-Reporting* können Sie die Geschäftsprozesse mit SAP BW/4HANA aufgrund der engen Verzahnung mit dem BI Content effizient weiterentwickeln.

### Merkmale und Kennzahlen

Bestes Beispiel hierfür sind Merkmale und Kennzahlen. Merkmale fassen die Stammdaten, Texte, Attribute von Attributen sowie Hierarchien über deren Schlüssel zentral zusammen und erzwingen damit ihre Harmonisierung. Wenn Sie entscheiden, dass ein Stammdatum zeitabhängig ist, wird sich diese Zeitabhängigkeit durch alle Datenmodelle durchziehen. So kann kein Entwickler, der mit den zugrundeliegenden Geschäftsprozessen wenig vertraut ist, diese Zeitabhängigkeit vergessen oder einfach ignorieren. Werden diese Merkmale zudem zur Filterung in eingeschränkten Kennzahlen verwendet, ist durch die Harmonisierung der Stammdaten gleichzeitig ein konsistentes Reporting gewährleistet.

Mit SAP BW/4HANA fallen die InfoProvider, die durch die Konsolidierung obsolet geworden sind, endgültig weg. Dies betrifft insbesondere den InfoCube mit all seinen Facetten sowie das klassische DataStore-Objekt (DSO), die InfoSets, TransientProvider und MultiProvider. Ohne Echtzeit-InfoCubes muss auch die Planung in die neuen DataStore-Objekte (advanced, kurz aDSOs) verlagert werden. Dies ist bereits seit SAP BW 7.4 auf klassischen DSOs vom Typ »Direktes Schreiben« möglich (siehe auch Abschnitt 1.3, »Die Wegbereiter: SAP-BW-Releases und ihre Eigenschaften«).

## 2.4 Modellierungsobjekte für SAP BW/4HANA

In diesem Abschnitt zeige ich Ihnen, wie Sie die verschiedenen Objekte in der neuen Modellierungsumgebung anlegen. Die neue Umgebung unterscheidet sich stark von der gewohnten Umgebung in SAP BW. Außerdem gibt es nun weniger Modellierungsobjekte als im klassischen SAP-BW-System.

Durch die vereinfachte Modellierung sind neue Architekturen und Konzepte möglich, z. B. die Verlagerung der OLAP- und Planungsfunktionen in die Datenbank oder der Datenzugriff auf beliebigen Ebenen des Schichtenmodells. Da es sich bei SAP BW/4HANA um eine neu ausgerichtete Architektur handelt, sind die Funktionen der Vorgängerobjekte nicht eins zu eins übertragbar. Auch wurden neue Funktionen hinzugefügt, die erst mit SAP BW/4HANA genutzt werden können.

### 2.4.1 Verbindung zwischen SAP-BW/4HANA-Server und BW-Modellierungswerkzeugen herstellen

Wie Sie die BW-Modellierungswerkzeuge bzw. SAP-BW/4HANA-Modellierungswerkzeuge installieren und ein SAP-BW/4HANA-Projekt einrichten, lernen Sie in Kapitel 3, »Administration«. In der SAP Cloud Appliance Library (CAL) können Sie jedoch direkt und ohne vorherigen Installationsaufwand in die Arbeit mit SAP BW/4HANA einsteigen. Dazu verbinden Sie sich mit dem ebenfalls in der Cloud bereitgestellten Windows-Server über das Remote Desktop Protocol (RDP). Den Link **Verbinden** zu diesem Server finden Sie in der CAL (siehe Abbildung 2.5). Melden Sie sich anschließend als Administrator mit Ihrem Masterpasswort am Windows-Server an.

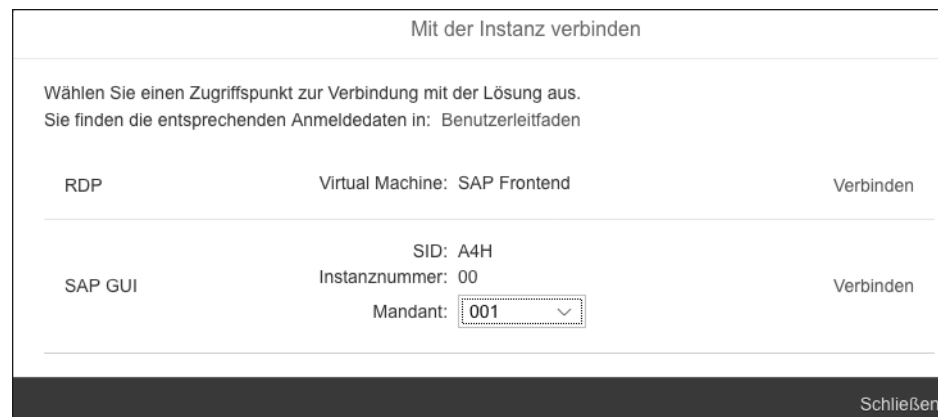


Abbildung 2.5 Verbindung mit dem Windows-Frontend-Server in der SAP Cloud Appliance Library

In diesem Abschnitt beschränke ich mich zunächst auf die Beschreibung des Verbindungsaufbaus mit dem SAP-BW/4HANA-Server über die BW-Modellierungswerkzeuge, so dass ich Ihnen im Anschluss die Anlage der SAP-BW/4HANA-Modellierungsobjekte vorführen kann.

Im SAP HANA Studio wird die Perspektive **BW Modeling** (die BW-Modellierungswerkzeuge) mit dem SAP-BW/4HANA-Server verbunden, wie in Abbildung 2.6 veranschaulicht.

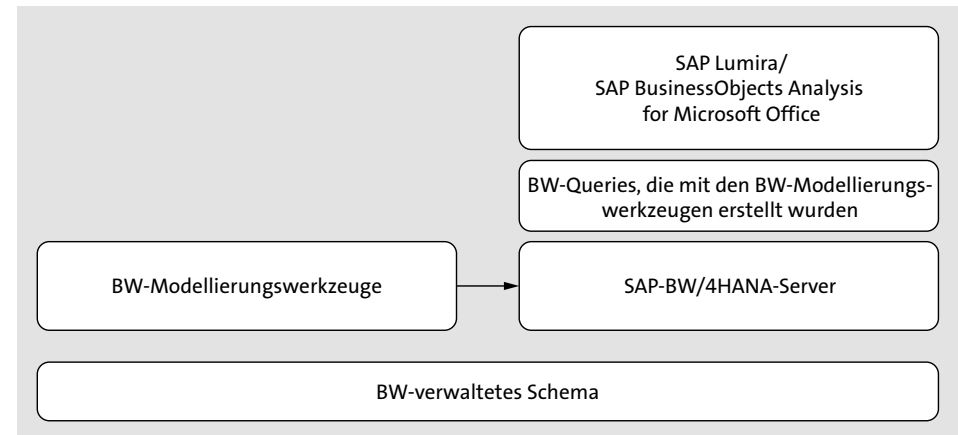


Abbildung 2.6 BW-veraltetes Schema

Nachdem Sie sich über das Remote Desktop Protocol mit dem Windows-Server verbunden haben bzw. Eclipse und die BW-Modellierungswerkzeuge erfolgreich installiert haben, können Sie die Verbindung zu Ihrer SAP-BW/4HANA-Instanz herstellen:

- Öffnen Sie hierfür die SAP-HANA-Administrationskonsole. Sie können diese Perspektive auf verschiedenen Wegen öffnen:
  - Sie wählen im Hauptmenü von Eclipse den Eintrag **Windows • Perspective • Open Perspective • Other** und scrollen dann in dem sich öffnenden Fenster herunter bis zum Eintrag **SAP HANA Administration Console**.
  - Sie können dieses Fenster auch direkt öffnen, indem Sie ganz rechts im Hauptmenü auf den Button **Open Perspective** (🏠) klicken. Je nach Anzahl der in Eclipse installierten Modellierungswerkzeuge kann diese Liste recht lang werden. Ich bevorzuge daher die folgende, dritte Option.
  - Links vom Button **Open Perspective** (🏠) befindet sich ein Eingabefeld mit dem Text **Quick Access**. Darüber starten Sie eine Suche nach dem eingegebenen Begriff. Geben Sie hier »SAP« ein, um alle SAP-relevanten Perspektiven und Kommandos zu sehen, wie in Abbildung 2.7 gezeigt.
- Die Perspektive **SAP HANA Administration Console** öffnet sich. Da Sie wahrscheinlich an einer ganzen Reihe paralleler Entwicklungen arbeiten werden, legen Sie am besten gleich zu Beginn ein Entwicklungsverzeichnis an. In diesem Buch ist der Auftraggeber die Redalli-Holding-Gesellschaft, kurz Redalli. Klicken Sie daher auf den Button **New** (📁) ganz links im Hauptmenü, und wählen Sie den Eintrag **Folder**. In das Feld **Folder Name** geben Sie »Redalli« ein und bestätigen mit **Finish** (siehe Abbildung 2.8).

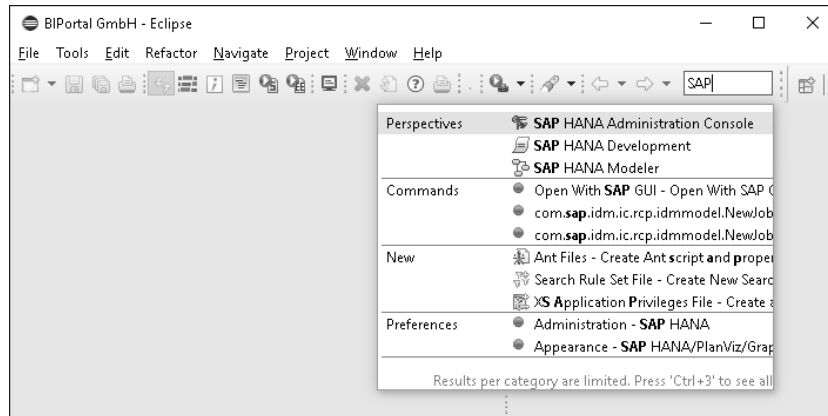


Abbildung 2.7 Auswahl der SAP-Administrationskonsole im SAP HANA Studio



Abbildung 2.8 Anlegen eines Projektordners im SAP HANA Studio

3. Ganz links unter **Systems** haben Sie nun die Möglichkeit, die Verbindung mit einem SAP-HANA-System herzustellen. Möchten Sie jedoch, dass Ihre Verknüpfungen dem korrekten Projektordner hinzugefügt werden, klicken Sie mit der rechten Maustaste auf den Projektordner. Über das sich öffnende Kontextmenü können Sie nun die Verbindung zu Ihrem SAP-BW/4HANA-System herstellen (siehe Abbildung 2.9).
4. Im folgenden Fenster geben Sie die Verbindungsparameter Ihrer SAP-BW/4HANA-Installation in den Amazon Web Services (AWS) an (siehe Abbildung 2.10). Dies sind die externe IP-Adresse des Linux-Servers, auf dem die SAP-HANA-Datenbank läuft, sowie dessen Instanznummer. Die IP-Adresse können Sie einfach aus dem Bereich **Informationen** Ihrer SAP BW/4HANA-Instanz kopieren (siehe Abschnitt 1.6.1, »SAP Cloud Appliance Library«). Laut Benutzerleitfaden ist die Instanznummer der SAP-HANA-Datenbank »02«.

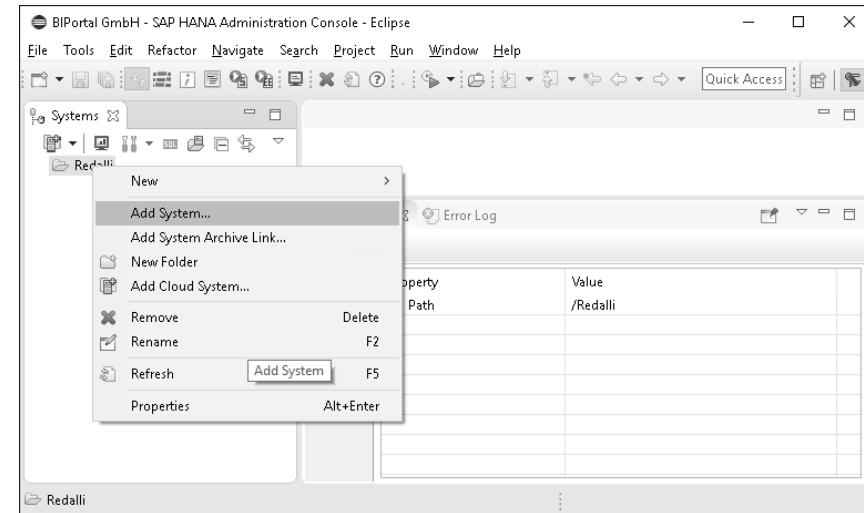


Abbildung 2.9 Verknüpfung eines Systems im SAP HANA Studio

### Schrägstrich im Produktnamen

Bei der Systembeschreibung sollten Sie bei jeder Gelegenheit auf den Schrägstrich im Namen »SAP BW/4HANA« verzichten, da er auf UNIX-Ebene als Zeichen für Dateipfade reserviert ist (siehe SAP-Hinweis 2354199).

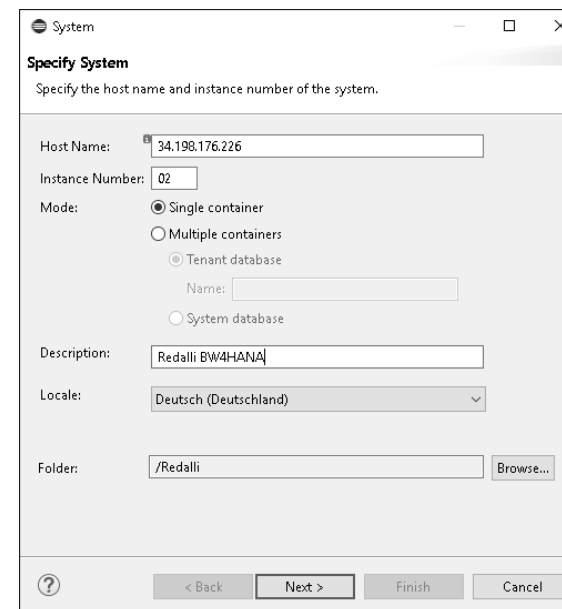


Abbildung 2.10 Verbindungsparameter hinzufügen

5. Als Modus wählen Sie **Single container**, da das SAP-HANA-System ausschließlich Ihnen gehört.



### Multiple Containers

Die Option **Multiple containers** ist insbesondere für Ihre Testinstallation auf der SAP Cloud Platform interessant (siehe Abschnitt 6.2.2, »SAP HANA Smart Data Integration«). Testkonten auf der SAP Cloud Platform teilen sich ein SAP-HANA-System, so dass Sie in diesem Fall spezifizieren müssten, mit welcher **Tenant database** Sie sich (quasi als Untermieter) verbinden möchten.

6. Bestätigen Sie nun Ihre Eingaben mit **Next**, um im letzten Bild die Kennung des Systembenutzers **SYSTEM** Ihrer SAP-HANA-Datenbank einzugeben (siehe Abbildung 2.11).

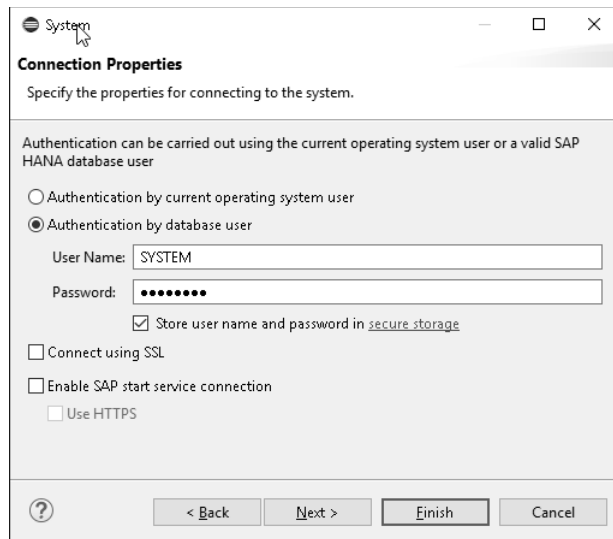


Abbildung 2.11 Verbindungseigenschaften zu SAP HANA anlegen

7. Setzen Sie auch das Häkchen bei **Store user name and password in secure storage**, um später nicht immer wieder die Systemkennung eingeben zu müssen. Die übrigen Checkboxen lassen Sie einfach frei.
8. Schließen Sie die Anlage der Verbindung mit **Finish** ab.

Anschließend erstellen Sie eine Verbindung mit dem SAP-BW/4HANA-Server:

1. Dazu bleiben Sie in der Perspektive **SAP HANA Administration Console** und wählen **New • Project**.
2. In dem folgenden Dialog wählen Sie den Wizard zur Anlage eines BW-Projekts (siehe Abbildung 2.12).

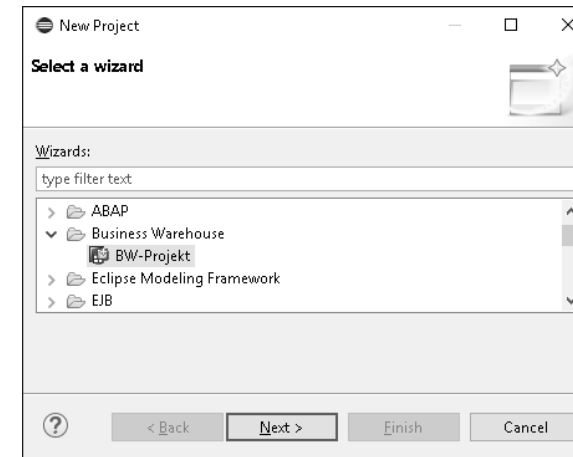


Abbildung 2.12 Neues BW-Projekt anlegen

3. Im folgenden Bild wählen Sie **New System Connection** (bzw. **Systemverbindung für das neue BW-Projekt**) und tragen die System-ID sowie die externe IP-Adresse des SAP-BW/4HANA-Servers ein (identisch mit den Daten des SAP-HANA-Servers, siehe Abbildung 2.13). Bestätigen Sie dann mit **Next**.

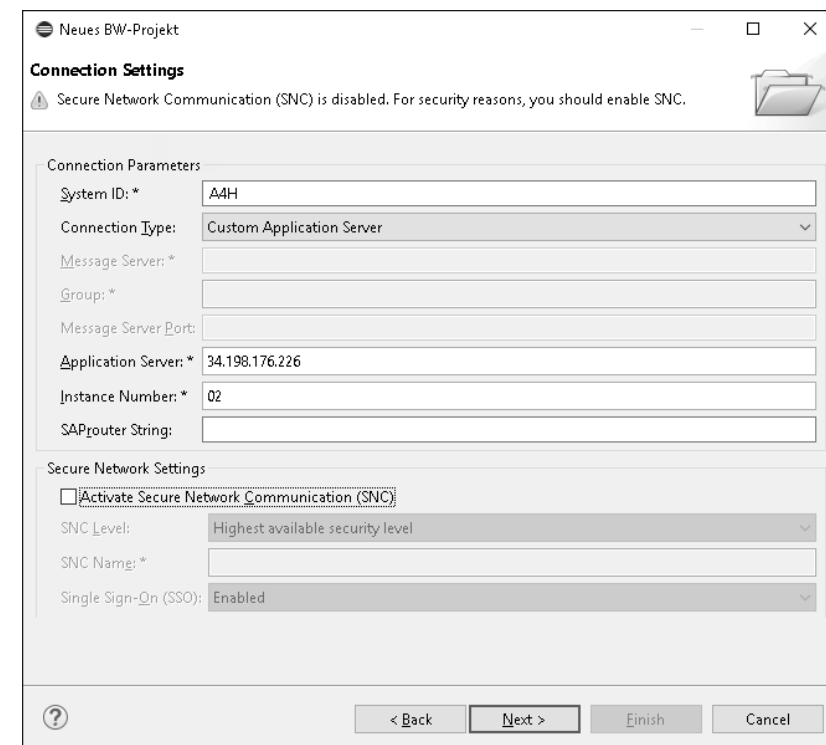


Abbildung 2.13 Neue SAP-BW/4HANA-Systemverbindung im SAP HANA Studio einrichten

4. Tragen Sie den Entwicklungsbutzer `DEVELOPER` ein, und bestätigen Sie mit **Finish** (siehe Abbildung 2.14).

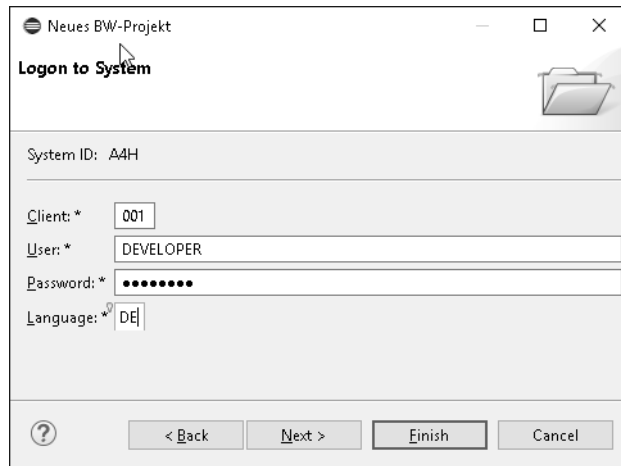


Abbildung 2.14 Entwicklungsbutzer für das SAP BW/4HANA-System



#### Eigenen BW-Projektnamen im Wizard vergeben

Alternativ können Sie mit einem weiteren Klick auf **Next** auch einen eigenen BW-Projektnamen vergeben. Dies hat den Vorteil, dass Ihre Projekte keinen generierten Namen nach dem Muster `<System-ID><Mandant><Benutzer><Anmeldesprache>` erhalten. Vergeben Sie eigene BW-Projektnamen, um sie insbesondere von Ihren zukünftigen ABAP-Projekten unterscheiden zu können. Die Einrichtung einer Verbindung mit dem verschlankten Application Server ABAP als SAP-BW/4HANA-Server zeige ich Ihnen in Abschnitt 4.5.1, »Projekt in den ABAP Development Tools anlegen«.

Mit diesen Verbindungen können Sie nun Ihren ersten Datenfluss erstellen; dies zeige ich Ihnen schrittweise in den folgenden Abschnitten.

#### 2.4.2 InfoArea anlegen

InfoAreas dienen wie in früheren SAP-BW-Releases der Gruppierung Ihrer Entwicklungsobjekte. Es steht Ihnen völlig frei, wie Sie Ihre SAP-BW/4HANA-Objekte ordnen. Häufig wird auf erster Ebene thematisch gruppiert. Auf den darunter liegenden Ebenen können Sie dann die Schichtenarchitektur Ihrer Implementierung abbilden. Im Fall von Redalli wird zwischen der Bonebene (Bon Layer), der Verteilungsebene für aufbereitete Daten (Propagation Layer) und der Analyseebene (Analytics Layer) unterschieden.

Zum Anlegen einer InfoArea gehen Sie wie folgt vor:

1. Wählen Sie im Kontextmenü Ihres BW-Repository bzw. in der übergeordneten InfoArea den Eintrag **New • InfoArea**.
2. In dem sich öffnenden Dialogfenster vergeben Sie einen technischen Namen sowie eine Beschreibung (siehe Abbildung 2.15).

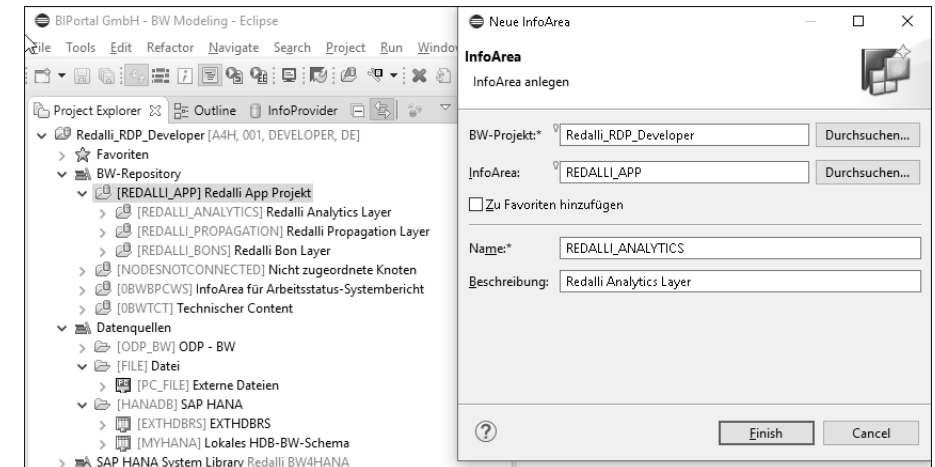


Abbildung 2.15 InfoArea in SAP BW/4HANA anlegen

3. Bestätigen Sie die Anlage der InfoArea mit **Finish**.

Wiederholen Sie diese Schritte für jede der im Project Explorer in Abbildung 2.15 gezeigten InfoAreas.

#### 2.4.3 Quellsystem anlegen

Um Quellsysteme im SAP HANA Studio anzubinden, müssen Sie zunächst eine Verbindung mit Ihrem SAP-BW/4HANA-System auf der Ebene des SAP-BW/4HANA-Servers herstellen:

1. Wechseln Sie hierzu in der Perspektive **BW Modeling** auf die Registerkarte **Project Explorer**, und melden Sie sich an Ihrem SAP-BW/4HANA-System an. Alternativ können Sie den Project Explorer auch über die Perspektive **SAP HANA Development** erreichen.
2. Unterhalb des Ordners **Datenquellen** sehen Sie in Abbildung 2.16 alle bestehenden ODP-, Datei- und SAP-HANA-Datenquellen (ODP\_BW, ODP\_SAP etc., FILE und HANADB). ODP steht hier für Operational Data Provisioning.



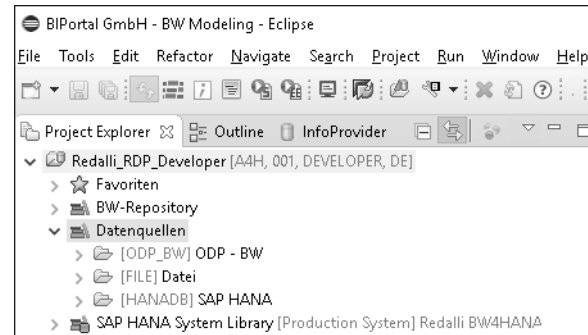


Abbildung 2.16 Anzeige der SAP-BW/4HANA-Datenquellen im SAP HANA Studio

Ab SAP BW/4HANA 1.0 SPS 02 wird es möglich sein, Quellsysteme über einen Wizard im SAP HANA Studio anzulegen, ohne dabei in den SAP GUI abspringen zu müssen. Neben den hier gezeigten Quellsystemen ODP, Datei und lokales SAP-HANA-Schema werden dann zusätzlich die Quellsystemtypen *SAP HANA Smart Data Access (SDA)*, *SAP-HANA-Tenant-Schema* (siehe Abschnitt 3.3, »Installation der Client-Werkzeuge für SAP HANA«) und *Big Data* (d. h. eine Spark-SQL-Destination) unterstützt. Diese Konzepte werden in Kapitel 6, »Datenintegration«, ausführlich erläutert.

Es kann sinnvoll sein, für die Anwendungskomponenten Ihrer zukünftigen DataSources eine ähnliche Struktur zu wählen, falls die Daten aus Ihrem SAP-BW/4HANA-System an andere Konsumenten weitergegeben werden sollen. In Abschnitt 6.1, »Anbindung der SAP-BW/4HANA-Quellsysteme«, zeige ich Ihnen, wie Sie Daten mithilfe von ODP-Funktionsbausteinen ausgehend von der Propagation Layer oder auch direkt aus der SAP-BW/4HANA-DataSource weiterleiten können.

Den Wizard zum Anlegen neuer Anwendungskomponenten erreichen Sie in den BW-Modellierungswerkzeugen über **File • New • Other • Business Warehouse • Application Component**. In dem sich öffnenden Dialog erhalten Sie jedoch aktuell (Version 1.17.7 auf SAP BW/4HANA 1.0 SPOO) die Meldung, dass im SAP-BW/4HANA-Projekt »Redalli RDP Developer« keine Anwendungskomponenten angelegt werden können. Zur Definition der Anwendungskomponenten-Hierarchie wechseln Sie daher in den SAP GUI, indem Sie auf den Button **Open SAP GUI** (🖱️) in der Navigationsleiste klicken (siehe Abbildung 2.17). Geben Sie den Transaktionscode RSA6 (DataSource und Hierarchie nachbearbeiten) ein. Wählen Sie hier **Knoten anlegen** (F6), und bilden Sie die Hierarchie Ihrer InfoAreas nach.

In den folgenden Abschnitten finden Sie ein Beispiel für jeden der in Abbildung 2.16 dargestellten Quellsystemtypen.

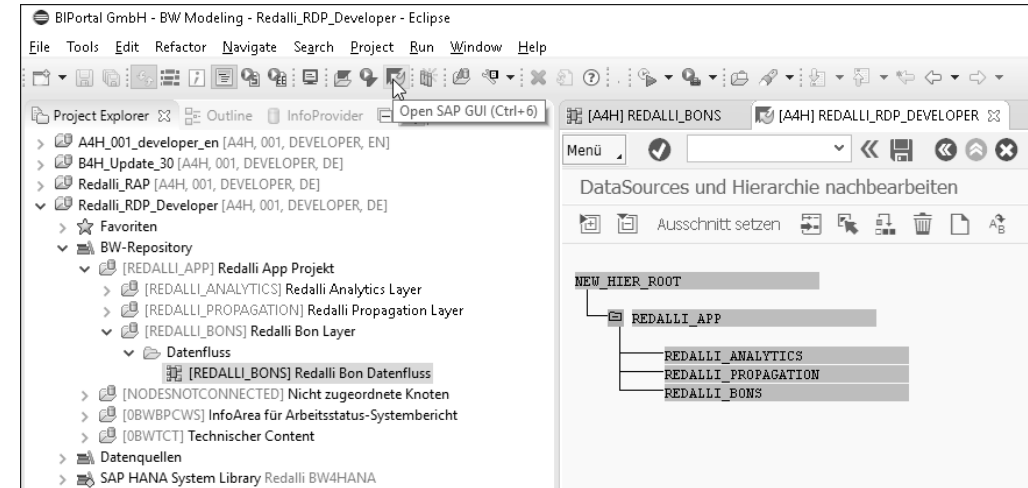


Abbildung 2.17 Anwendungskomponenten in SAP BW/4HANA anlegen

### Operational Data Provisioning

Der Quellsystemtyp Operational Data Provisioning (ODP) kombiniert die Fähigkeiten der Quellsystemtypen SAPI-Extraktor, SAP BW, Webservices und ODP, die in früheren SAP-BW-Releases zur Verfügung standen. Zusätzlich ermöglicht er die direkte Anbindung der folgenden Quellsysteme:

- SAP HANA Information Views (ODP\_HANA)
- SLT-Queues (SAP Landscape Transformation, ODP\_SLT)
- ABAP-CDS-Views (Core Data Services, ODP\_CDS)
- SAP Business ByDesign (SAPs SaaS-Lösung für den Mittelstand, ODP\_BYD)

Insgesamt gibt es in SAP BW/4HANA also sieben Typen von ODP-Quellsystemen; diese sind in Abbildung 2.18 zu sehen.

Quellsysteme	tech. Name
• ODP - BW	ODP_BW
• ODP - SAP (Extraktoren)	ODP_SAP
• ODP - SAP-HANA-Informationssicht	ODP_HANA
• ODP - SLT-Queue	ODP_SLT
• ODP - ABAP CDS Views	ODP_CDS
• ODP - SAP Business ByDesign	ODP_BYD
• ODP - Andere Kontexte	ODP
> Datei	FILE
> SAP HANA	HANADB

Abbildung 2.18 Quellsystemtypen in SAP BW/4HANA

Das ODP-Framework existiert bereits seit SAP BW 7.4 SPS05 vollumfänglich. In produktiven SAP-BW-Systemen wird es jedoch bis heute noch nicht durchgängig eingesetzt. Das mag daran liegen, dass ODP in früheren SAP-BW-Releases optional war. Mit

SAP BW/4HANA wird diese Technologie verpflichtend. Daraus ergeben sich für Sie einige wichtige Vorteile:

- Sie verzichten auf die Zwischenspeicherung in der PSA. Dadurch gewinnen Sie wertvolle Zeit beim täglichen Laden.
- ODP-DataSources greifen direkt auf die *Operational Delta Queue* (ODQ) im Quellsystem zu. Dadurch haben Sie Zugriff auf aktuelle Daten, die Sie gegebenenfalls mit einem Open ODS View basierend auf einer ODP-DataSource direkt in einer Query in Analysis for Office anzeigen können.
- Die ODQ kann mehrere Konsumenten gleichzeitig bedienen. Es findet also kein Klonen der Delta-Queues für die einzelnen DataSources der verschiedenen SAP-BW-Systeme mehr statt. Zusammen mit der Komprimierung der ODQ im Quellsystem bedingt dies einen wesentlich geringeren Speicherbedarf für Delta-Queues.

Die ODQ können Sie in der Transaktion ODQMON im ODQ-Monitor des Quellsystems einsehen. Das Anlegen neuer ODP-Quellsysteme erfolgt über das SAP-GUI-Fenster im SAP HANA Studio oder direkt im SAP GUI. Ihr SAP-BW/4HANA-System können Sie sehr einfach selbst als ODP-Quellsystem anlegen, wie in Abbildung 2.19 gezeigt. Das funktioniert auch, wenn Ihr SAP-BW/4HANA-System in einer Cloud wie den AWS gehostet wird.

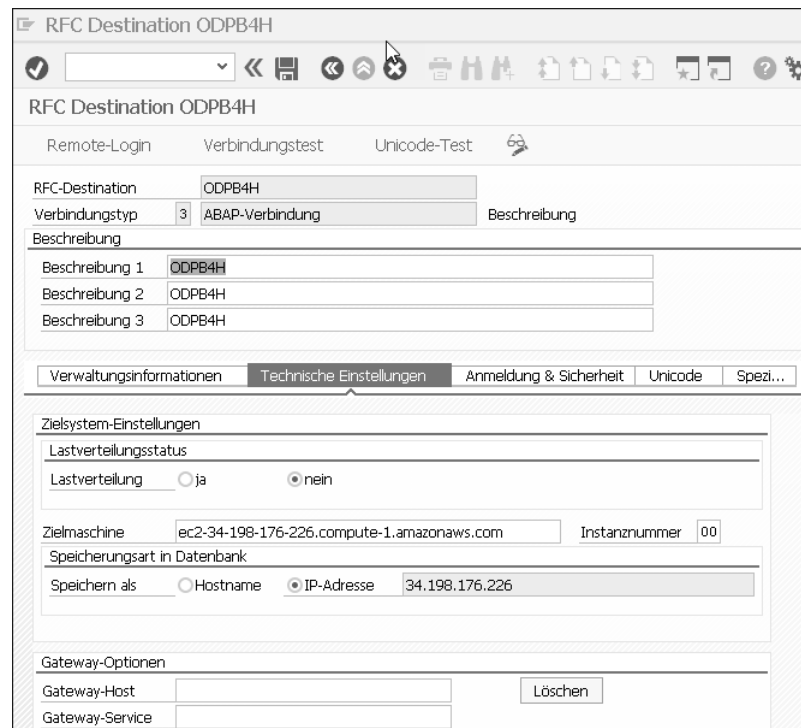


Abbildung 2.19 Eigenes SAP-BW/4HANA-System als ODP-Quelle anlegen

In der CAL verwenden Sie hierzu den Benutzer BWREMOTE mit dem Passwort icecream bzw. mit dem Passwort aus dem Benutzerleitfaden Ihres CAL-Accounts. Anschließend sichern und testen Sie die Verbindung.

### Zugriff auf das lokale SAP-HANA-Datenbankschema

Der Quellsystemtyp HANADB vereint die Quellsystemtypen DB Connect, SAP Data Services, BAPI/Partner (Drittanbieter) und SAP-HANA-Quellsysteme früherer SAP-BW-Releases. Ich zeige Ihnen in diesem Abschnitt, wie Sie eine externe Instanz der SAP-HANA-Datenbank als Quellsystem Ihrer SAP-BW/4HANA-Installation anbinden können.

Das lokale BW-Schema der SAP-HANA-Datenbank ist in Ihrem SAP-BW/4HANA-Testsystem in der CAL bereits als MYHANA eingetragen (siehe Abbildung 2.20).

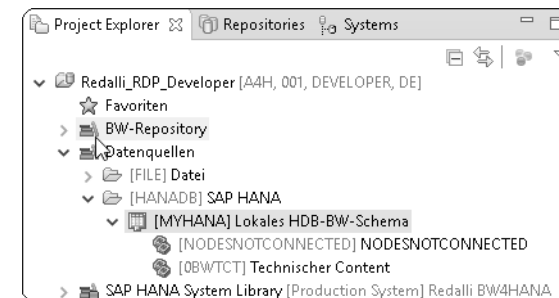


Abbildung 2.20 Lokales HDB-BW-Schema im Quellsystem MYHANA

Aus Sicht des SAP-BW/4HANA-Systems ist dies gleichzeitig das einzig relevante SAP-HANA-Datenbanksystem in der skalierbaren Architektur für Business Intelligence. Besteht in Ihrer Architektur die Notwendigkeit, weitere SAP-HANA-Systeme direkt anzuschließen, müssen diese entfernten SAP-HANA-Systeme als sogenannte *Remote DataSources* über SAP HANA Smart Data Access angebunden werden. Dies beschreibe ich in Kapitel 6, »Datenintegration«.

### Dateien

Das Anlegen einer FILE-DataSource, d. h. einer Datei als Datenquelle, funktioniert wie in früheren SAP-BW-Releases, da sich hierbei wieder der SAP GUI öffnet (siehe Abbildung 2.21).

Im rechten Fenster öffnet sich der SAP GUI mit dem bekannten Dialog **DataSource ändern**. In Abbildung 2.21 lese ich beispielsweise eine Datei mit Daten zu den mit der Redalli-Karte gesammelten Punkten ein, die in den weiteren Abschnitten dieses Kapitels wiederverwendet wird. Diese Datei können Sie von der Website des Verlages unter [www.sap-press.de/4384](http://www.sap-press.de/4384) im Bereich **Materialien zum Buch** herunterladen.

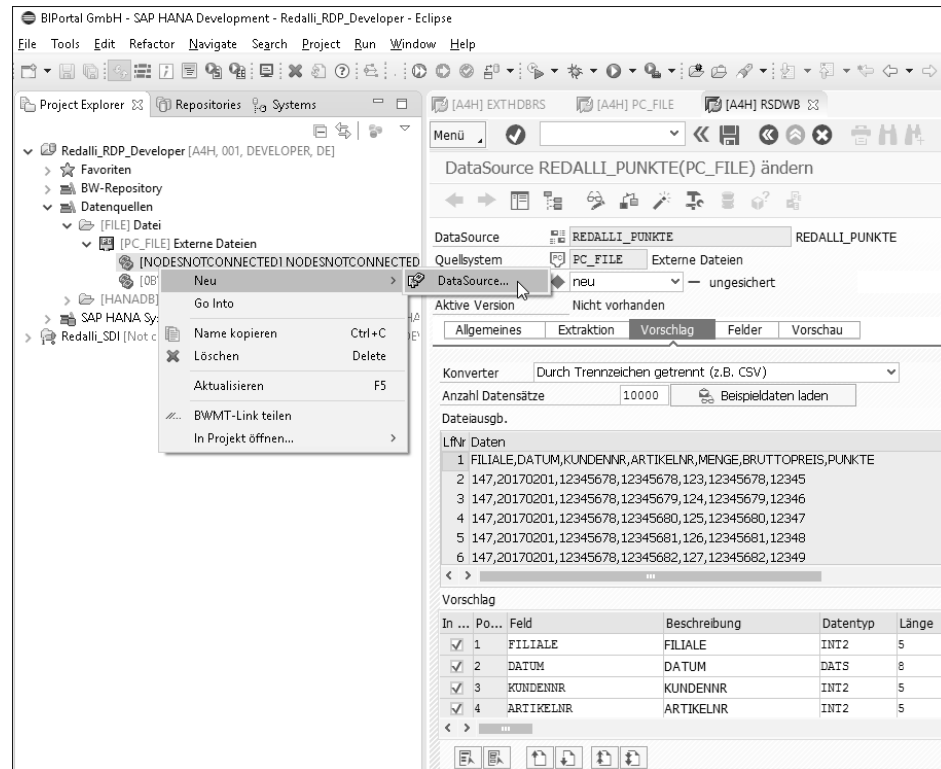


Abbildung 2.21 DataSource vom Typ PC\_FILE anlegen

#### 2.4.4 Datenfluss anlegen

SAP BW/4HANA erlaubt es Ihnen, den geplanten Datenfluss bereits anzulegen, bevor die darin enthaltenen Objekte angelegt wurden. Dies gibt Ihnen als zukünftigem SAP-BW/4HANA-Architekten die Möglichkeit, die geplante Architektur frühzeitig mit den Ansprechpartnern des Projekts zu besprechen, ohne abstrakte Dokumente anlegen zu müssen.

1. Navigieren Sie hierzu zur gewünschten InfoArea. In deren Kontextmenü wählen Sie den Eintrag **Neu • Datenfluss** (siehe Abbildung 2.22).
2. In der **Palette** auf der rechten Seite finden Sie die verfügbaren Objekte für Datenflüsse (siehe Abbildung 2.23). Wenn Sie Erfahrung mit früheren SAP-BW-Releases gesammelt haben, werden Sie diese Objekte sofort wiedererkennen:
  - DataSources
  - InfoSources
  - Open-Hub-Destinationen
  - InfoObjects

- DataStore-Objekte (advanced)
- Aggregationsebenen
- Open ODS Views
- CompositeProvider

Das Anlegen all dieser Objekte wird in den folgenden Abschnitten anhand eines einfachen Beispiels erläutert.

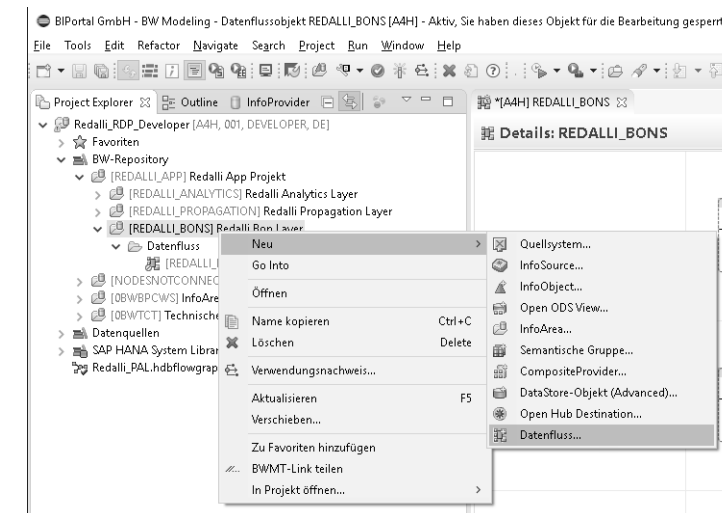


Abbildung 2.22 Datenfluss in SAP BW/4HANA anlegen

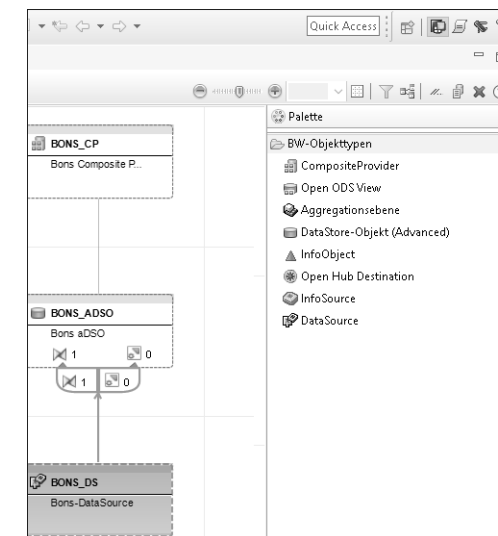


Abbildung 2.23 Palette der verfügbaren Objekte

- Für den Datenfluss der Bonebene ziehen Sie einfach die Objekte **DataSource**, **DataStore-Objekt (Advanced)** und **CompositeProvider** auf die Designfläche links neben der Palette.
- Anschließend verbinden Sie diese Objektschablonen des Datenflusses mithilfe der Drag-&Drop-Funktion mit der linken Maustaste.
- Für jedes Objekt geben Sie einen technischen Namen sowie eine Beschreibung an. Markieren Sie dazu als erstes Ihr aDSO im Designbereich, dann wechseln Sie im Fenster **Properties** auf die Registerkarte **Documentation** (siehe Abbildung 2.24).

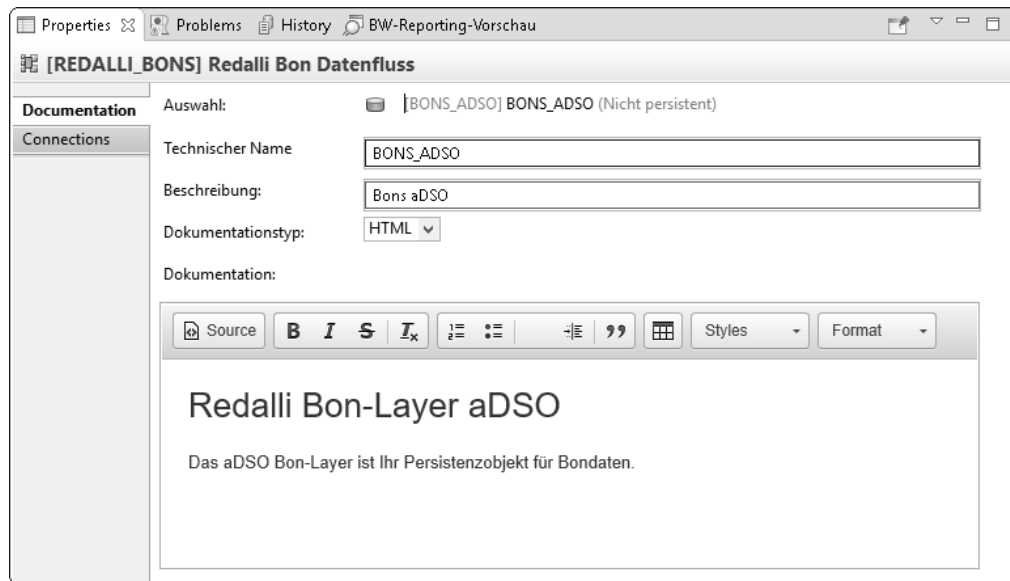


Abbildung 2.24 Dokumentation eines DataStore-Objekts (advanced) in SAP BW/4HANA

- Sie können dem aDSO eine ausführlichere Dokumentation hinzufügen, als es das einzeilige Beschreibungsfeld zulässt. Dazu wählen Sie den **Dokumentationstyp** »HTML«, der Ihnen zahlreiche Formatierungsmöglichkeiten eröffnet. Falls Ihnen die Funktionen der Format-Buttons nicht ausreichen, können Sie mit dem Button **Source** auf der Registerkarte **Documentation** in den HTML-Code abspringen und beliebige HTML-Anweisungen einfügen.
- Verfahren Sie mit den anderen Objekten genauso.



#### Tutorium für HTML

Eine Übersicht von HTML-Anweisungen finden Sie unter:  
<https://www.w3schools.com/html>

- Überprüfen Sie auf der Registerkarte **Connections** die angelegten Verbindungen (siehe Abbildung 2.25). Die Übersicht zeigt, dass die in der Designsicht gezogenen Verbindungslinien registriert wurden. Da die Objekte noch nicht physisch existieren, werden sie mit der Bemerkung **(Nicht persistent)** angezeigt.
- Aktivieren Sie anschließend Ihren Datenfluss mit der Tastenkombination **[Strg] + [F3]** bzw. mit einem Klick auf den Button **BW-Objekt aktivieren** in der Funktionsleiste.

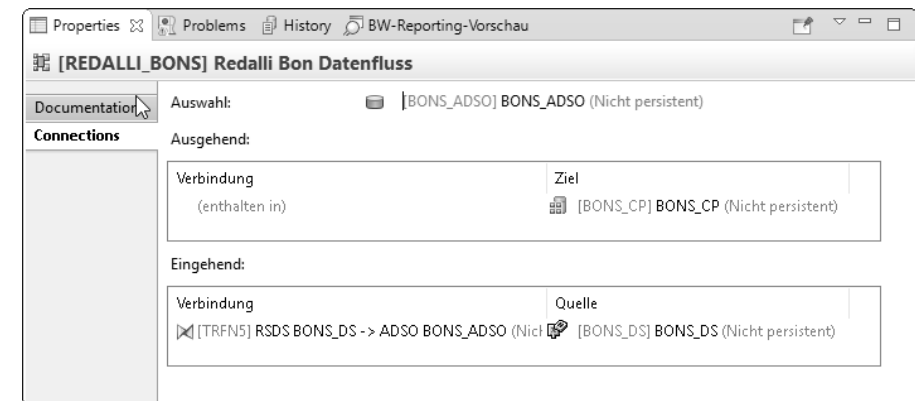


Abbildung 2.25 Eingehende und ausgehende Verbindungen eines DataStore-Objekts (advanced) in SAP BW/4HANA

- Wechseln Sie nun auf die Registerkarte **Problems** neben der Registerkarte **Properties**, um den Erfolg der Aktivierung zu prüfen. Wie in Abbildung 2.26 sollten vier Warnungen angezeigt werden, da Sie vier Objekte vorbereitet haben: Neben Ihren drei benannten Objekten wurde auch eine Transformation eingefügt. Im Abschnitt **Infos** sehen Sie, dass Ihr Datenfluss konsistent und valide aktiviert wurde.

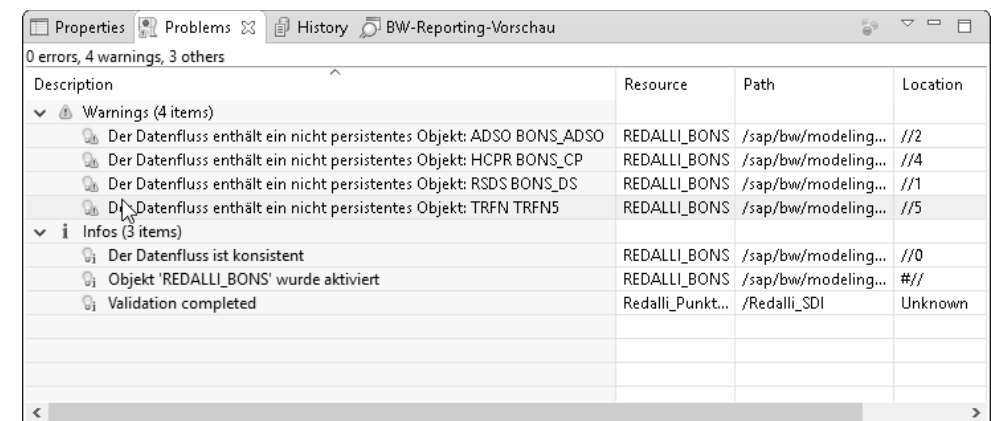


Abbildung 2.26 Aktivierungsprotokoll eines Datenflusses in SAP BW/4HANA

Die Objekte Ihres Datenflusses können Sie zudem geordnet in der **Outline**-Sicht Ihres SAP-BW/4HANA-Projekts sehen. Wechseln Sie hierzu in der Perspektive **BW Modeling** auf die Registerkarte **Outline** (siehe Abbildung 2.27).

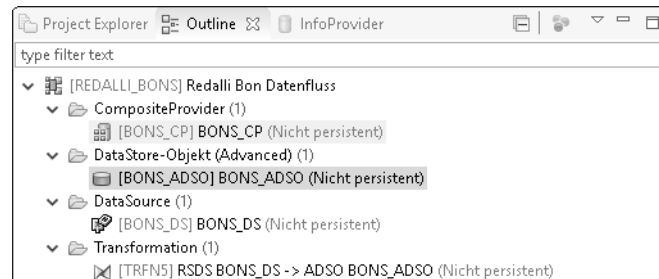


Abbildung 2.27 Outline-Sicht eines Datenflusses in SAP BW/4HANA

Diese Übersicht ist insbesondere für komplexe Datenflüsse mit zahlreichen Objekten nützlich. In der Funktionsleiste zum Datenfluss stehen Ihnen zudem weitere hilfreiche Funktionen wie die komprimierte Sicht (📄), das Ausblenden von Datentransferprozessen oder Transformationen (🔍) sowie das automatische Layout (📐) zur Verfügung (siehe Abbildung 2.28). Die Historie auf der Registerkarte **History** zeigt Ihnen die Zeitstempel der aktiven (A) und zuletzt modifizierten (M) Version Ihres Datenflusses an.

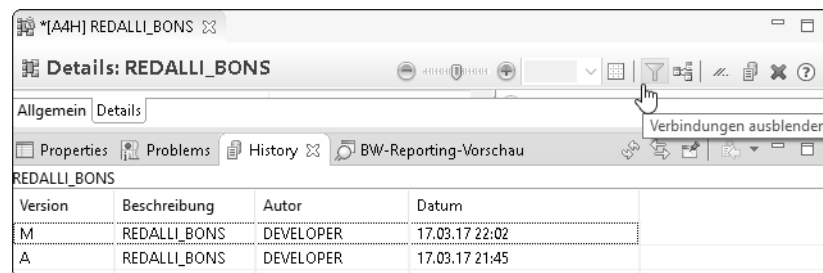


Abbildung 2.28 History-Sicht eines Datenflusses in SAP BW/4HANA

Die M-Version bedeutet, dass der Datenfluss in Bearbeitung und daher für Änderungen durch andere Entwickler gesperrt ist. In der Titelleiste von Eclipse werden Sie daran entsprechend erinnert. Nach ihrer Aktivierung verschwindet die M-Version aus der Historie, die Sperrung wird aufgehoben und die A-Version erhält den Zeitstempel der letzten Aktivierung.

### 2.4.5 Open ODS View anlegen

Mit SAP BW/4HANA ist die Zeit des logischen Data Warehousing gekommen. Dank der Technologie SAP HANA Smart Data Integration ist es nun leichter, unterschied-

liche Datenquellen zu integrieren, ohne sie replizieren zu müssen. So können schnell erste Ergebnisse erzielt werden. Eine fehlende Datenreplikation kann jedoch zu erhöhter Latenz im Reporting führen. Gründe hierfür sind die Engpässe Netzwerk und Verbindungsaufbau zum Quellsystem sowie das fehlende Caching. Der *Open ODS View* kann auf Datenquellen zugreifen, deren Strukturen wie eine SAP-HANA-Datenbanktabelle durch Felder und Feldtypen definiert sind. Dies sind neben den SAP-HANA-Datenbanktabellen auch Views, die gegebenenfalls über eine Remote-Source-Anbindung auf externe Datenquellen zugreifen.

#### Operational Data Store

ODS steht für den *Operational Data Store*. Die Open ODS Views werden also als Views für den Zugriff auf operative Daten in Echtzeit modelliert (siehe auch W.H. Inmon, »Building the Operational Data Store«, John Wiley & Sons 1996).

Über *Calculation Views* in der SAP-HANA-Datenbank können zudem komplexe Vorberechnungen erfolgen. Diese können Sie über die generierten Column Views den Open ODS Views zur Verfügung stellen (siehe Abschnitt 4.2.5, »Open ODS Views auf Column Views«). Ebenso können Sie Daten aus dem SAP-BW/4HANA-verwalteten Datenbankschema mit anderen SAP-HANA-Schemata verbinden. Innerhalb von SAP BW/4HANA können auch DataSources mit Direktzugriff auf Daten als Quelle dienen.

Neben diesen Möglichkeiten gibt es allerdings auch zahlreiche Einschränkungen der Funktionalität der Open ODS Views gegenüber den persistenten Objekten in SAP BW/4HANA, den InfoObjects und den aDSOs. So können Open ODS Views zwar in Queries verwendet werden, ihre Felder werden jedoch mit einem generierten technischen Namen nach dem Muster 4<InfoProvider>-<Feld-Name> angezeigt. Stammdaten wie Attribute und Hierarchien zu den Feldobjekten können nicht verwendet werden. Für Kennzahlen können keine Ausnahmeaggregationen definiert werden.

Zu diesen funktionalen kommen auch technische Einschränkungen hinzu. Beispielsweise kann der OLAP-Cache nicht für Queries genutzt werden. Daher stellt sich die Frage, weshalb Open ODS Views überhaupt verwendet werden sollten. Aus der Sicht eines SAP-BI-Architekten haben Open ODS Views einen Vorteil bei der Erstellung von BI-Modellen. Ändern sich die Bewegungsdaten innerhalb eines Tages und müssen den Entscheidern sofort zugänglich gemacht werden (wie beispielsweise Bestandsdaten einem Filialleiter), haben Sie keine andere Wahl, als die Daten direkt aus der Quelle abzugreifen.

Mit den Open ODS Views können Sie – ohne InfoObjects und nur auf den Feldern der Quelle basierend – Daten live in SAP BW/4HANA bereitstellen. Zwar könnten Sie das auch mit einem CompositeProvider realisieren, jedoch zeige ich Ihnen in Abschnitt 4.2.3, »CompositeProvider auf Open ODS Views«, dass die Zwischenschaltung eines Open ODS View wesentlich komfortabler und robuster ist. Ein wesentlicher Grund

hierfür ist die Konvertierung der SAP-HANA-Datentypen in die kompatiblen Datentypen des ABAP Dictionary (DDIC), die Grundlage der InfoObjects sind, die Sie in Ihren Queries verwenden sollten.

Ein auf dem Open ODS View basierender CompositeProvider sorgt zudem für die volle OLAP-Funktionalität, die Sie allein mit dem Open ODS View nicht hätten. Composite-Provider können allerdings nur auf anderen InfoProvidern oder auf SAP HANA Information Views aufsetzen. Open ODS Views können hingegen auf DataSources, SAP-HANA-Tabellen und -Views, aDSOs, Transformationen (über die Auswahl der Quelle und des Transformationsziels) sowie auf virtuellen Tabellen aufsetzen, die Sie über SAP HANA Smart Data Access aus entfernten Systemen referenzieren können. Die Handhabung dieser *Remote Sources* zeige ich Ihnen in Abschnitt 6.1.2, »SAP HANA Smart Data Access«.

Voraussetzung für die Anlage eines Open ODS View ist die Verknüpfung Ihres SAP-BW/4HANA-Projekts mit der zugrundeliegenden SAP-HANA-Datenbank. Wählen Sie dazu im Kontextmenü Ihres Projekts den Eintrag **SAP-HANA-System zuordnen**. Im folgenden Dialog wählen Sie Ihre SAP-HANA-Verbindung aus und bestätigen mit **Finish**.

### Open ODS View für eine DataSource anlegen

Legen Sie nun wie im Folgenden beschrieben einen Open ODS View für die in Abschnitt 2.4.3, »Quellsystem anlegen«, erzeugte FILE-DataSource mit den Punkteständen der Redalli-Kundenkarten an:

1. Dazu wählen Sie im Kontextmenü Ihrer InfoArea **Redalli Bon Layer** den Eintrag **Neu • Open ODS View**.
2. Im sich öffnenden Dialog vergeben Sie einen technischen Namen entsprechend Ihrer SAP-BW/4HANA-Namenskonvention.



#### Namenskonventionen für Objekte in SAP BW/4HANA erstellen

Da es zahlreiche mögliche Datenquellen gibt, bietet es sich an, z. B. den Typ DataSource als »DS« abzukürzen oder »TAB« als Abkürzung für Tabellen zu nutzen. Beachten Sie dabei, dass der technische Name auf mindestens drei und maximal zwölf Stellen beschränkt ist.

3. Anschließend fügen Sie eine Beschreibung hinzu.
4. Im Bereich **Semantik** können Sie wählen, ob es sich bei der Datenquelle um Transaktionsdaten (**Fakten**), oder um **Stammdaten** bzw. **Texte** handelt (siehe Abbildung 2.29). In unserem Beispiel sind es Transaktionsdaten, die aus der Nutzung der Redalli-Kundenkarte stammen.

5. Im Abschnitt **Quelltyp** wählen Sie entsprechend **DataSource (BW)**. Die übrigen Optionen sind:
  - **Datenbanktabelle oder -view** für HDB-Objekte
  - **Virtuelle Tabelle mit SAP HANA Smart Data Access** für referenzierte externe Tabellen (*Remote Source Table*)
  - **DataStore-Objekt (Advanced)** für aDSOs
  - **Transformation** für die Extraktstruktur einer Transformation. Da diese Struktur kein SAP-BW/4HANA-Objekt ist, müssen Sie hierfür die DataSource als Quelle der Transformation bzw. eine InfoSource als Ziel der Transformation auswählen.
6. Wählen Sie anschließend **Next**.

Abbildung 2.29 Open ODS View anlegen

7. Auf dem folgenden Bildschirm wählen Sie Ihre DataSource aus (siehe Abbildung 2.30). Geben Sie dazu im Suchfenster einfach einen Stern (\*) ein, um sich alle DataSources anzeigen zu lassen. Wählen Sie Ihre DataSource aus, und bestätigen Sie anschließend mit **Finish**.

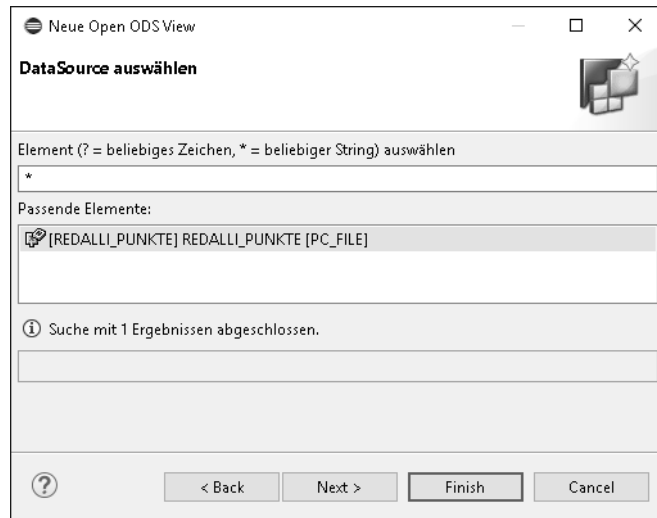


Abbildung 2.30 DataSource beim Anlegen eines Open ODS View auswählen

8. In der folgenden Übersicht erhalten Sie einen Link, der auf Ihre DataSource verweist und Sie direkt in die Data Warehousing Workbench führt. Den Quelltyp können Sie nachträglich über **Bearbeiten** anpassen. Über **Datenfluss generieren** können Sie Ihren Open ODS View in eine Ladestrecke mit dem Zieltyp InfoSource oder aDSO einbinden. Bei den verwendeten Datentypen haben Sie die Möglichkeit, die **Quellobjekttypen** oder die entsprechenden **BW-Datentypen** zu verwenden.
9. Zunächst möchten Sie jedoch einen ersten Blick auf die Daten werfen. Bestätigen Sie also mit **Next**.
10. Wechseln Sie dann auf die Registerkarte **Fakten** (siehe Abbildung 2.31). Im linken Fenster sehen Sie die **Quellfelder** mit ihren Datentypen, Feldlängen und Dezimalstellen. Außerdem ist hier festgehalten, ob die Groß- und Kleinschreibung für die Felder relevant ist. Mögliche Konvertierungsroutinen entsprechend den Angaben aus Ihrer DataSource können Sie ebenfalls einsehen.

Daneben stehen die **Sicht-Felder**, also die sichtbaren Felder des Open ODS View. Ziehen Sie hier per Drag & Drop die Felder **DATUM**, **KUNDENNR** und **ARTIKELNR** auf den **Schlüssel**. Anschließend ziehen Sie die **FILIALE** in den Ordner **Merkmale** und fügen dem Ordner **Kennzahlen** die Spalten **BRUTTOPREIS**, **MENGE** und **PUNKTE** hinzu.

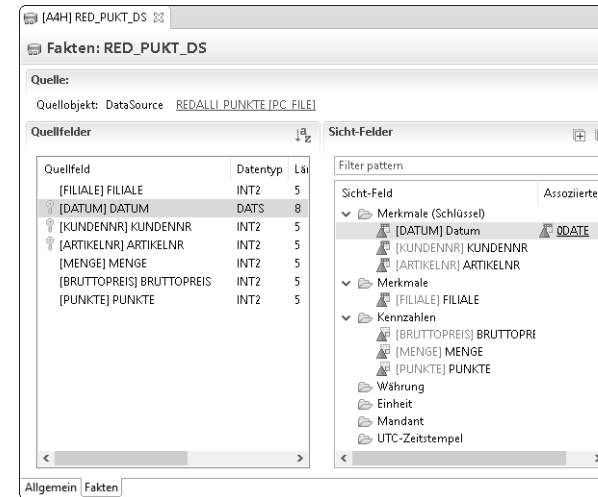


Abbildung 2.31 Auswahl der im Open ODS View sichtbaren Felder

### Assoziation anlegen

Beim Datum haben Sie die Möglichkeit, das Feld mit einem bekannten InfoObject, dem Merkmal ODATE, zu assoziieren. Öffnen Sie dazu auf der rechten Seite der Registerkarte **Fakten** im Fenster **Allgemein** den Bereich **Assoziation**. Dort wählen Sie über die Auswahlliste die Option **Mit InfoObject** und in dem Feld daneben das Merkmal ODATE (siehe Abbildung 2.32). ODATE hat keine verwendbaren Navigationsattribute und ist auch nicht an ein anderes Merkmal *gekennzeichnet*, so dass die Auswahl der Buttons **Navigationsattribute** bzw. **Klammerung** hier ein leeres Ergebnis zeigt.

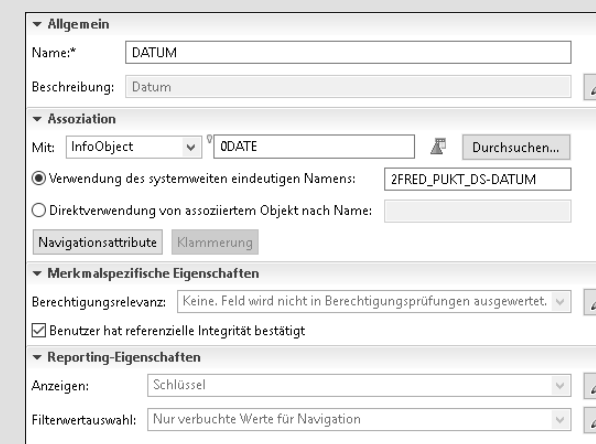


Abbildung 2.32 Fenster »Allgemein« auf der Registerkarte »Fakten« in der Data Warehousing Workbench

Im Bereich **Merkmalspezifische Eigenschaften** können Sie anmerken, dass der Benutzer, d. h. Sie als Entwickler, die referenzielle Integrität Ihrer Daten bestätigt. Inhaltlich bedeutet dies, dass die Datensätze Ihrer Quelle nur Einträge enthalten, die bei assoziierten Merkmalen den Stammdaten zugeordnet werden können. Dies sollte grundsätzlich bei assoziierten Merkmalen der Fall sein, da die Assoziierung sonst keinen Sinn ergibt.

ODATE hat keine **Berechtigungsrelevanz** für Redalli, so dass Sie die Grundeinstellung **Keine** stehen lassen können.

11. Abschließend prüfen Sie Ihre Einstellungen mit **BW-Objekt prüfen** (**Strg** + **F2**) und aktivieren Ihren Open ODS View (**Strg** + **F3**). Das Protokoll bestätigt Ihnen die erfolgreiche Prüfung und Aktivierung (siehe Abbildung 2.33).



Abbildung 2.33 Aktivierungsprotokoll eines Open ODS View

Nun können Sie Ihren angelegten Open ODS View ansehen. In der Funktionsleiste finden Sie oben rechts den Eintrag **InfoProvider-Sicht anzeigen** (🔍). Auf der Registerkarte **InfoProvider** erhalten Sie nun eine geordnete Übersicht Ihrer Merkmale und Kennzahlen (siehe Abbildung 2.34).

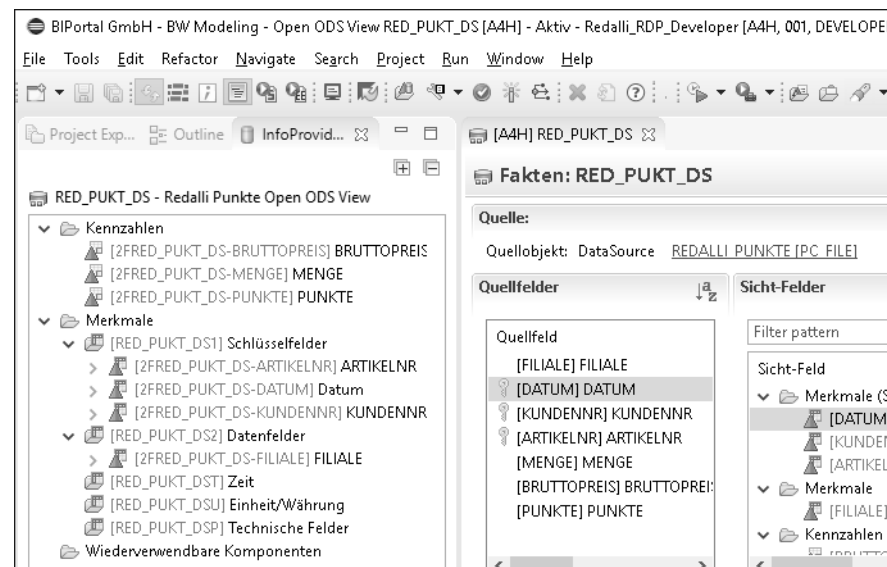


Abbildung 2.34 Übersicht der Merkmale und Kennzahlen eines Open ODS View

Wählen Sie die **BW-Reporting-Vorschau** im Menü unter **Navigate**. Im unteren Fenster öffnet sich die Datenvorschau mit zahlreichen Zusatzinformationen wie beispielsweise den **Systemdaten** (siehe Abbildung 2.35).



Abbildung 2.35 Reporting-Vorschau eines Open ODS View (Systemdaten)

### Open ODS View für eine native SAP-HANA-Tabelle anlegen

Wirklich interessant werden Open ODS Views in gemischten Szenarien, d. h. wenn Sie Daten aus anderen Schemata in SAP BW/4HANA verarbeiten möchten. Dies ist insbesondere dann der Fall, wenn Sie Data Science betreiben und beispielsweise eine Klassifikation Ihrer Kunden vornehmen. Dank einer solchen Klassifikation können Sie z. B. Muster in den Ernährungsstilen erkennen und so Vorhersagen bezüglich des Kaufverhaltens neuer Kunden treffen. Diese Algorithmen werden agil entwickelt, weshalb das Marketing sehr daran interessiert ist, sofort die Auswirkungen einer neuen Klassifikation auf die Bestandsdaten zu sehen. In Kapitel 8, »Advanced Analytics«, zeige ich Ihnen, wie Sie solch eine Kundensegmentierung vornehmen und das Ergebnis zur Vorhersage des Kaufverhaltens verwenden können. Dazu wird in Kapitel 8 eine native SAP-HANA-Tabelle mit den Kundendaten angelegt, die ich Ihnen hier als Open ODS View präsentieren möchte.

Einen Open ODS View für eine native SAP-HANA-Tabelle legen Sie im Grunde genauso an wie einen Open ODS View für eine DataSource. Allerdings müssen Sie dem Systembenutzer SAP<System-ID> (im Fall Redalli SAPA4H) zuvor Leserechte für das Schema mit den Bontabellen erteilen. Leserechte sind Teil der Privilegien und werden ausführlich in Abschnitt 3.4, »Sicherheitsarchitektur in SAP BW/4HANA«, besprochen. Als kurzen Vorgriff möchte ich Ihnen kurz die wichtigsten Schritte erläutern, um diesen Open ODS View anlegen zu können. Die Leserechte für den Systembenutzer Ihres SAP-BW/4HANA-Systems vergibt der Administrator, der in unserem Beispiel Inhaber des Schemas mit den Bondaten ist. Er öffnet im SAP HANA Studio seine SQL-Konsole und setzt folgenden Befehl ab:

```
grant select on schema BONS to SAPA4H;
```



Das genügt bereits, um einen Open ODS View auf Tabellen dieses Schemas anlegen zu können. Gehen Sie dazu wie folgt vor:

1. Wählen Sie im Kontextmenü Ihres neuen Ordners **Open ODS View** den Eintrag **Neu • Open ODS View**, und vergeben Sie wieder einen technischen Namen sowie eine Beschreibung (siehe Abbildung 2.36).

Abbildung 2.36 Open ODS View für eine externe SAP-HANA-Tabelle anlegen

2. Wählen Sie im Feld **Semantik** erneut **Fakten** aus, da es sich bei den Bons um Transaktionsdaten der Kassen handelt. Im Feld **Quellentyp** wählen Sie jetzt jedoch **Datenbanktabelle oder -view** aus. Schließlich wählen Sie die Option **Inhaber / Schema**, da sich die Bondaten im Schema BONS und nicht in einem bekannten Schema des SAP-BW/4HANA-Systembenutzers SAPA4H wie MYHANA befinden.

3. Bestätigen Sie nun mit **Next**. Geben Sie im Suchfeld einen Stern (\*) ein, um die Liste der verfügbaren Schemata zu erhalten (siehe Abbildung 2.37).

Abbildung 2.37 Schema auswählen

#### Berechtigung zur Anzeige der Schemata

Das Ihnen diese Schemata zur Auswahl angezeigt werden, sagt noch nichts über Ihre Leseberechtigungen für diese aus. Der SAP-BW/4HANA-Systembenutzer SAPA4H hat eine Leseberechtigung für die Metatabellen der SAP-HANA-Datenbank, so dass Sie die verfügbaren Schemata sehen können. Um aber eine Tabelle darin auslesen zu können, benötigen Sie die entsprechenden Privilegien, die im Vorfeld durch den Administrator zugeteilt wurden.

4. Im Folgedialog geben Sie erneut einen Stern (\*) ein, um alle Tabellen des Schemas BONS zu sehen (siehe Abbildung 2.38).
5. Wählen Sie hier die Tabelle REDALLI\_BONS, und bestätigen Sie mit **Next**.
6. Sie werden nun aufgefordert, ein neues Quellsystem vom Typ HANADB (SAP HANA) für das Schema BONS anzulegen. Verwenden Sie einen technischen Namen, der Ihrer Namenskonvention für Quellsysteme entspricht, und bestätigen Sie mit **Finish** (siehe Abbildung 2.39). Sie können auch zuvor mit **Next** noch einmal alle Angaben überprüfen und anschließend mit **Finish** bestätigen.

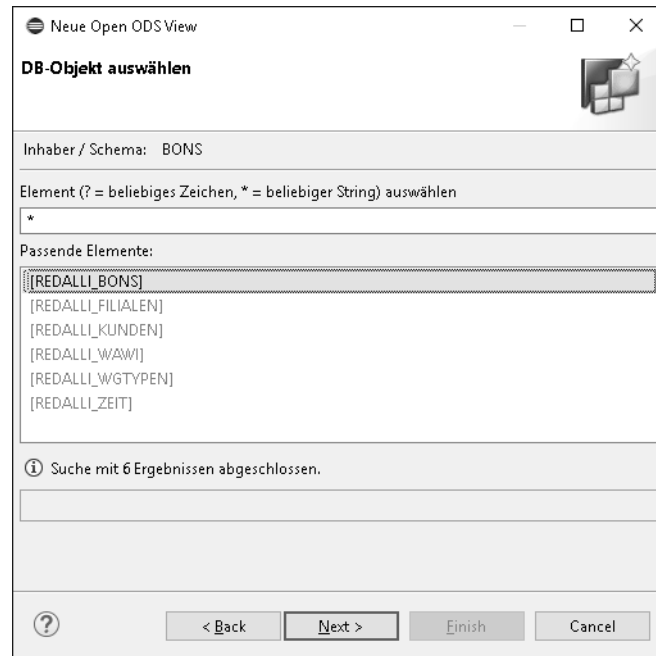


Abbildung 2.38 Tabellen des Schemas BONS auswählen

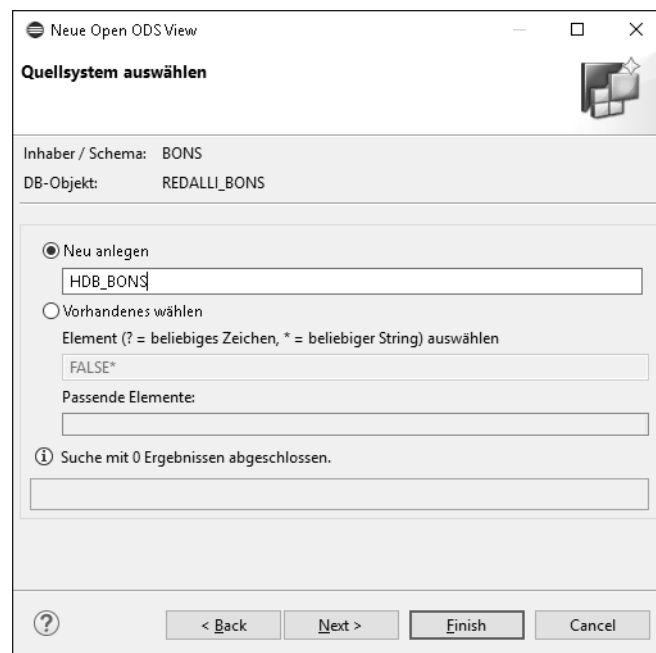


Abbildung 2.39 Ein neues SAP-HANA-Quellsystem für externe Tabellen im Open ODS View anlegen

### Bestehendes Quellsystem für weitere Open ODS Views

Für alle weiteren Tabellen dieses Schemas müssen Sie das Quellsystem nicht jedes Mal neu anlegen. Stattdessen wählen Sie für alle weiteren Open ODS Views die Option **Bestehendes Quellsystem** und geben im folgenden Suchfeld einen Stern (\*) ein (siehe Abbildung 2.40).

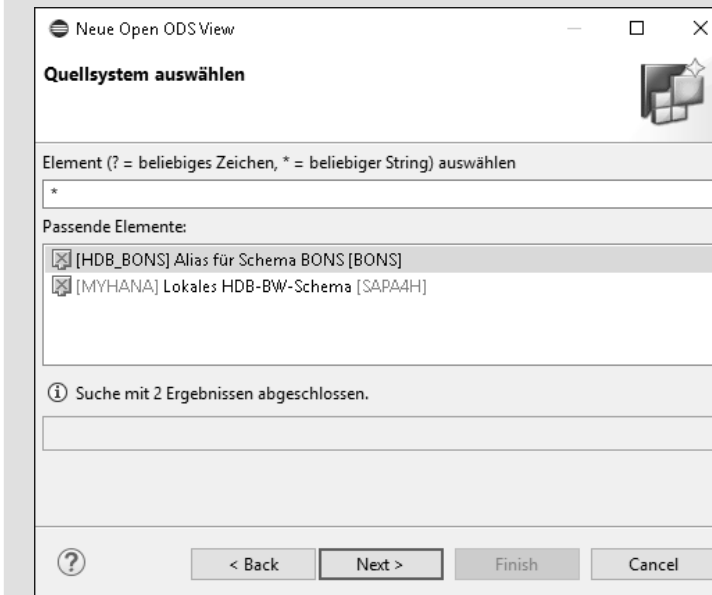


Abbildung 2.40 Bestehendes SAP-HANA-Quellsystem für Open ODS Views auswählen

7. Um den Open ODS View zu vervollständigen, wechseln Sie auf die Registerkarte **Fakten** im sich öffnenden Dialog **Fakten: RED\_BONS\_HT**. Falls Sie die Tabelle REDALLI\_BONS in Kapitel 8 korrekt angelegt haben, befinden sich im Ordner **Schlüssel** bereits alle Schlüsselfelder.
8. Fügen Sie hier noch dem Ordner **Merkmale** per Drag & Drop das Feld **KUNDENNR** hinzu. Dem Ordner **Kennzahlen** fügen Sie das Feld **MENGE** hinzu (siehe Abbildung 2.41).
9. Anschließend überprüfen Sie die Konsistenz Ihres Open ODS View mit **BW-Objekt prüfen** und aktivieren ihn über die Schaltfläche in der Funktionsleiste (🔧).

Nach erfolgreicher Aktivierung Ihres Open ODS View können Sie sich dessen Daten in der Datenvorschau anzeigen lassen (siehe Abbildung 2.42).

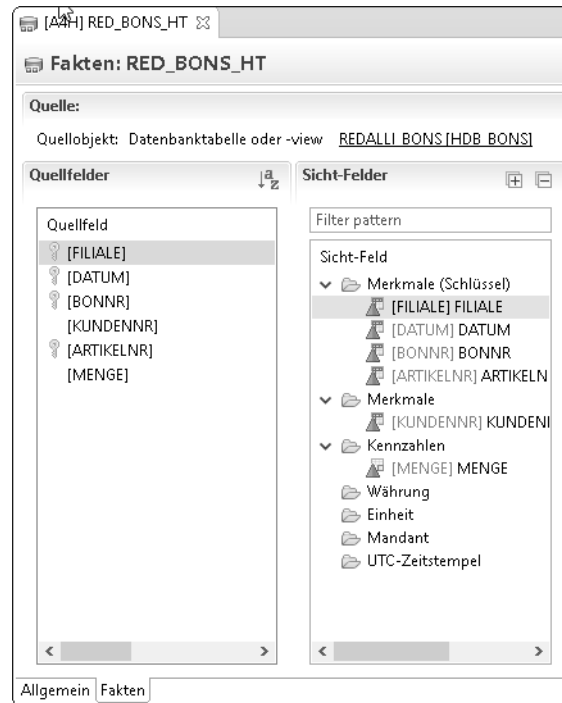


Abbildung 2.41 Externe SAP-HANA-Tabelle als Quelle eines Open ODS View

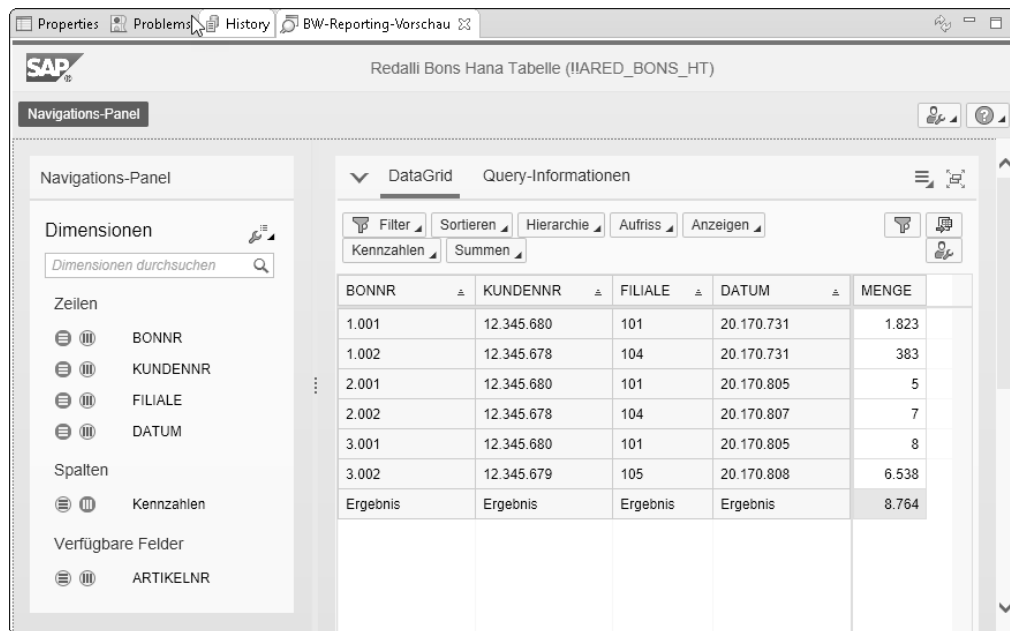


Abbildung 2.42 Vorschau der Bondaten des Open ODS View in SAP BW/4HANA

Über das **Navigations-Panel** können Sie dem Aufriss Merkmale und Kennzahlen hinzufügen und wieder löschen.

### 2.4.6 DataSource anlegen

Da das Marketing von Redalli mit den Daten zufrieden ist, beginnen Sie nun mit dem Aufbau Ihres Datenflusses und legen zunächst eine DataSource an:

1. Öffnen Sie Ihren Datenfluss REDALLI\_BONS, und wählen Sie im Kontextmenü des ersten Knotens BONS\_DS den Eintrag **Persistentes BW-Objekt anlegen** (siehe Abbildung 2.43).

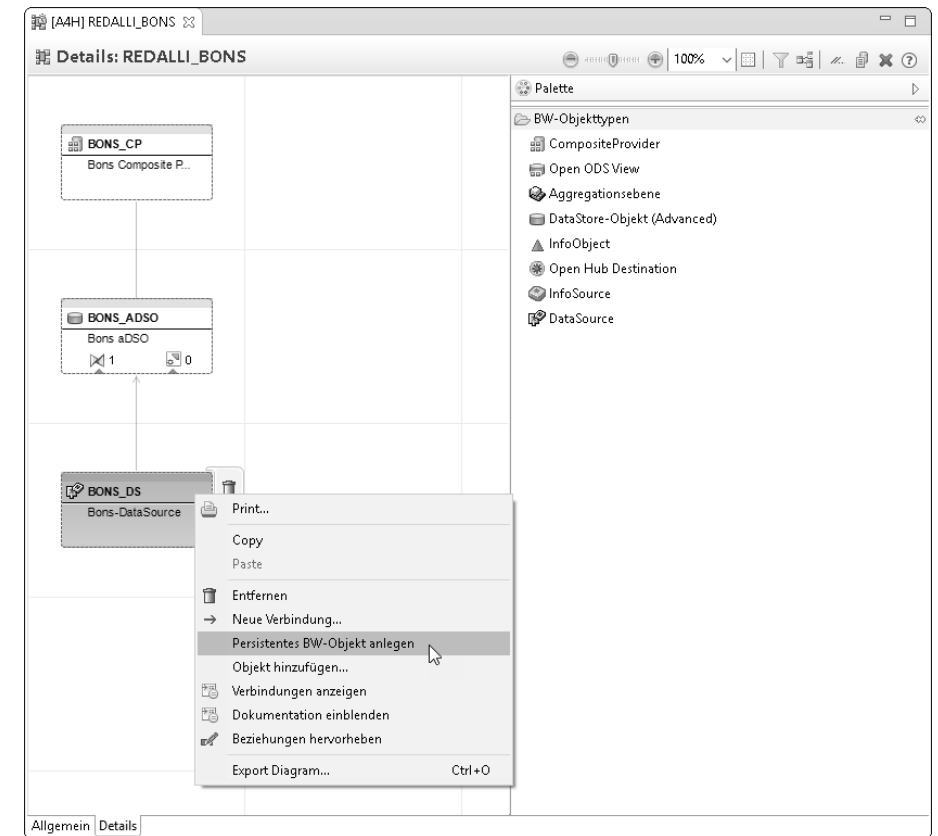


Abbildung 2.43 Persistieren der DataSource im Datenfluss

2. Im folgenden Dialog wählen Sie das von Ihnen zuvor angelegte Quellsystem HDB\_BONS aus (siehe Abbildung 2.44).