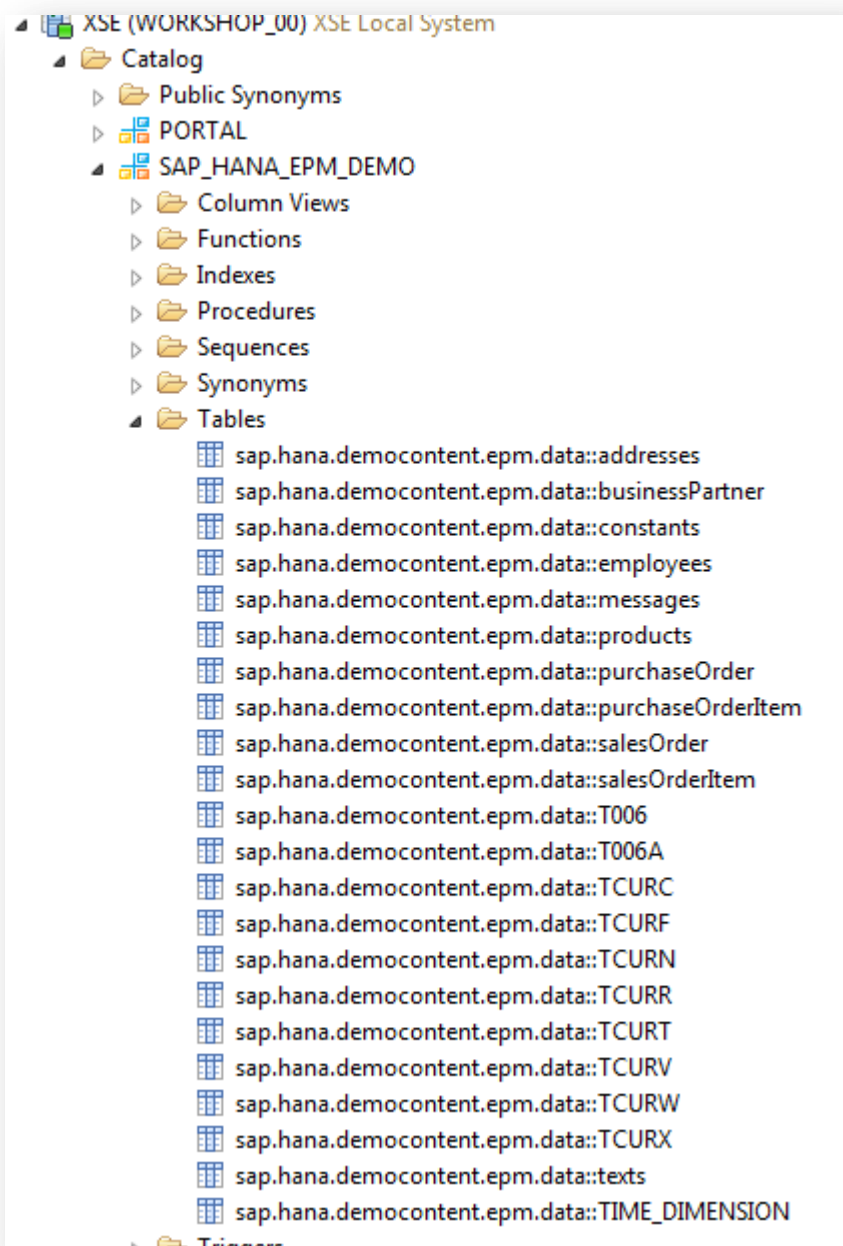


Exercise 2: Modeling Views

There are three types of views – **Attribute Views**, **Analytic Views**, and **Calculation Views**. In this exercise, we will utilize these modelling tools to create some more advanced views.

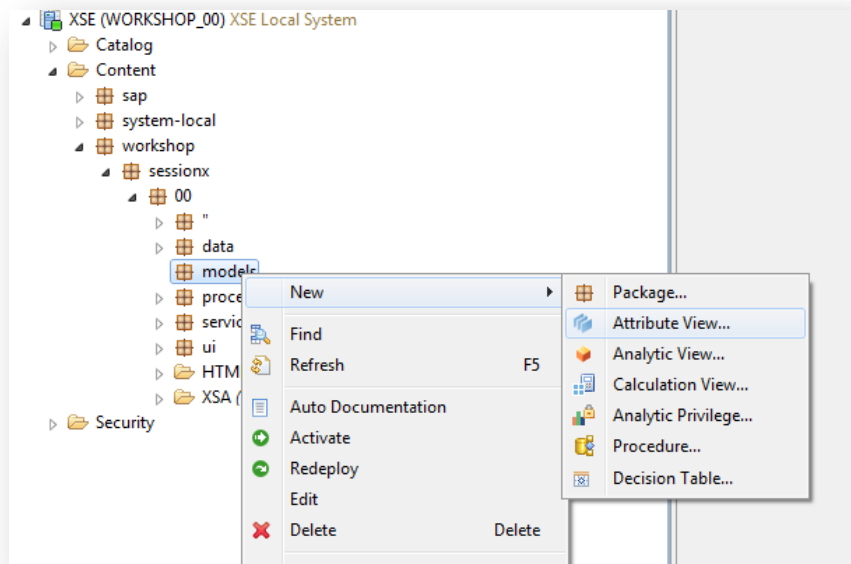
Rather than spending time creating a larger series of tables and sample data, the workshop will leverage the standard EPM (Enterprise Procurement Model) demo model. This model already exists in the workshop under the schema SAP_HANA_EPM_DEMO and contains Sales Orders, Purchase Orders, and supporting master data.

Please note that all the files, views which we will create in this exercise are case sensitive.

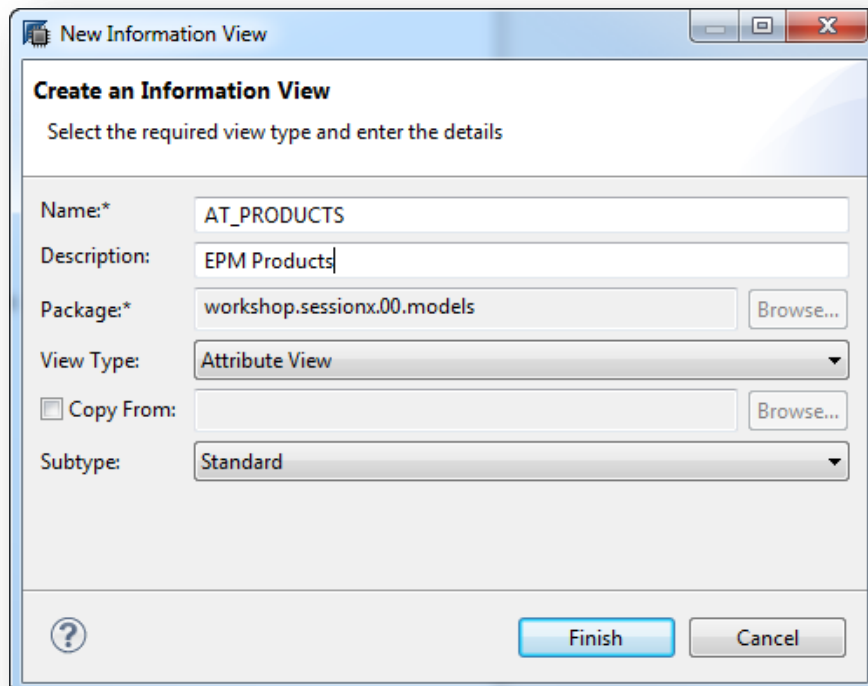


1. Attribute View

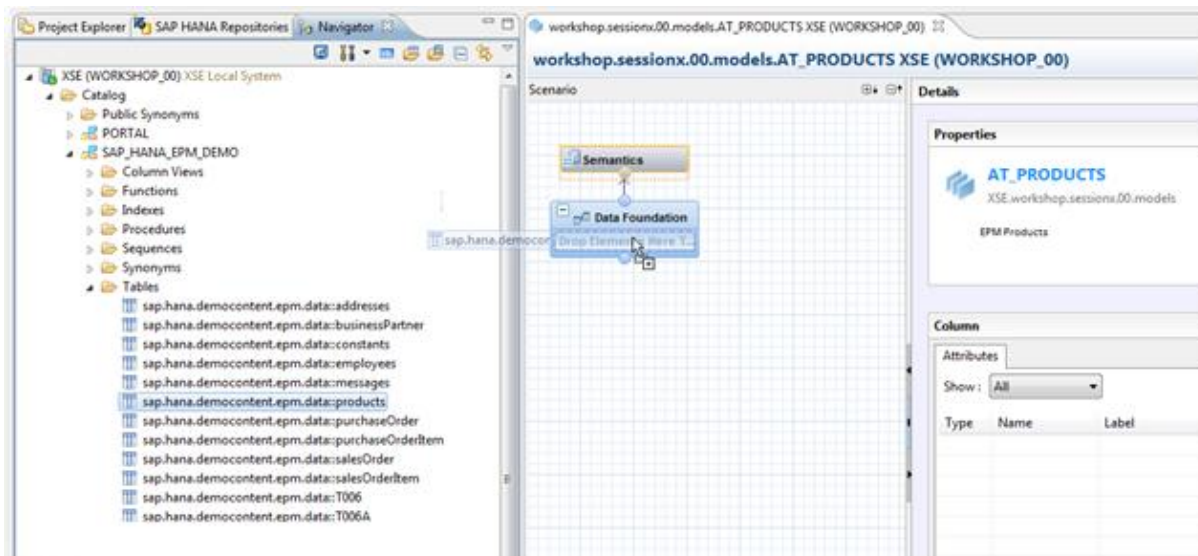
- For this exercise, you might want to switch to the **Modeler perspective** in SAP HANA Studio as we will be using the SAP HANA Systems view.
- In the models sub-package of your workshop package, create a new attribute view based upon the expanded information for the products. This will require joining the **products**, **texts**, **businessPartner**, and **addresses** tables.
- From the right mouse click on the models package, choose New->Attribute View



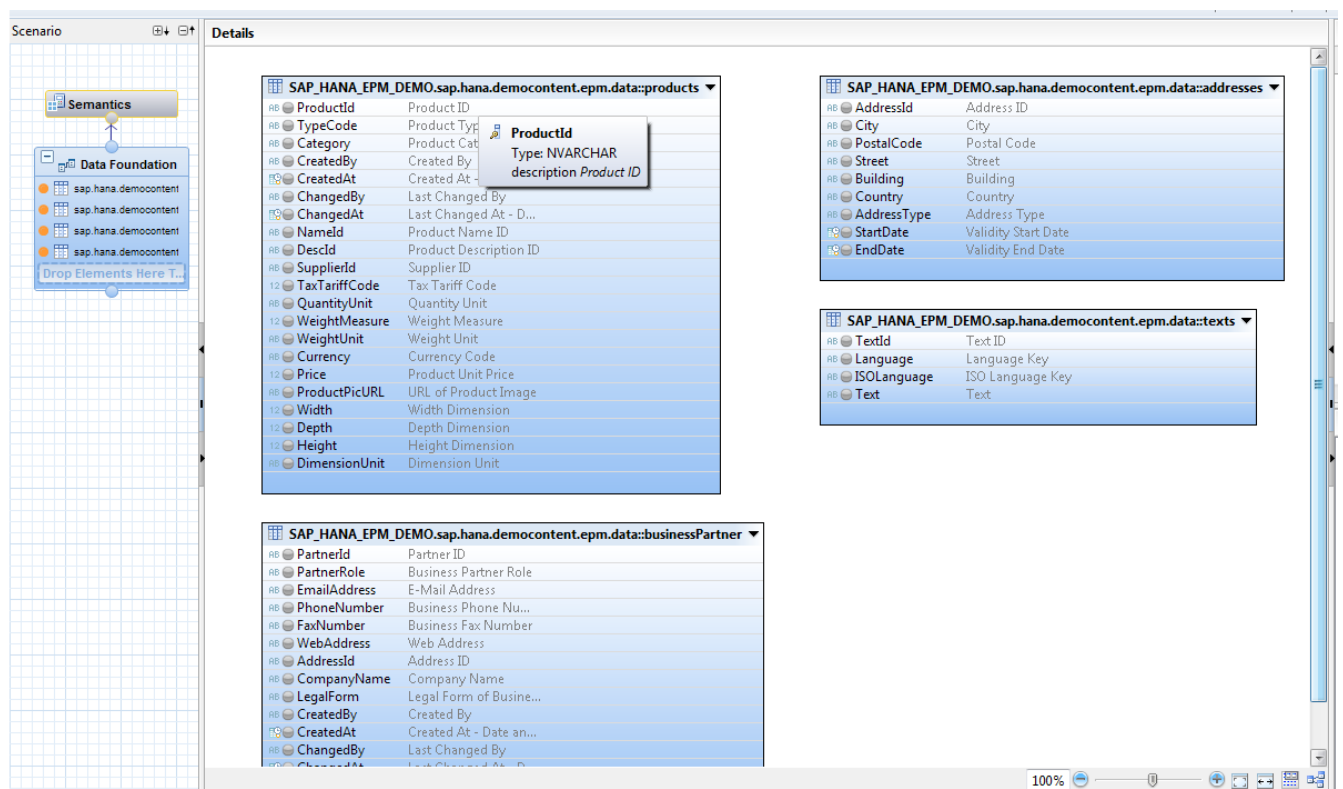
- Create an Attribute View named **AT_PRODUCTS**



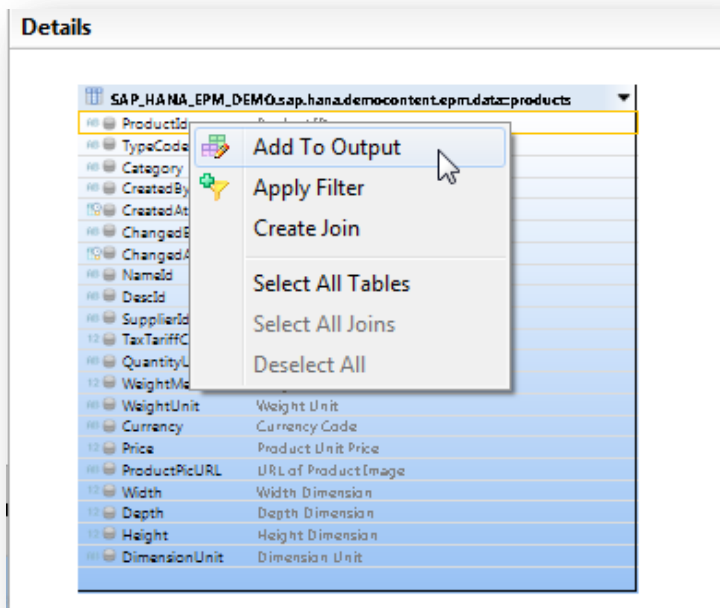
- e. To add tables to the data foundation of your view, you can drag and drop them from the SAP_HANA_EPM_DEMO Schema.



- f. Add **products**, **businessPartner**, **addresses** and **texts** to your data foundation.



- g. Using the Add to Output option from the context menu, add the **ProductId**, **TypeCode**, **Category**, **Currency**, and **Price** fields from the **products** table to the output.



Details

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:products ▼	
AB	ProductId Product ID
AB	TypeCode Product Type Code
AB	Category Product Category
AB	CreatedBy Created By
AB	CreatedAt Created At - Date an...
AB	ChangedBy Last Changed By
AB	ChangedAt Last Changed At - D...
AB	NameId Product Name ID
AB	DescId Product Description ID
AB	SupplierId Supplier ID
12	TaxTariffCode Tax Tariff Code
AB	QuantityUnit Quantity Unit
12	WeightMeasure Weight Measure
AB	WeightUnit Weight Unit
AB	Currency Currency Code
12	Price Product Unit Price
AB	ProductPicURL URL of Product Image
12	Width Width Dimension
12	Depth Depth Dimension
12	Height Height Dimension
AB	DimensionUnit Dimension Unit

- h. Optionally, you can change the name of a column as it becomes part of the view. For example you might change **Category** to **ProductCategory**.

The screenshot shows the SAP HANA Studio interface. On the left, the 'Scenario' pane shows a 'Data Foundation' project. The 'Details' pane in the center displays a list of columns for the 'SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data::products' table. The 'Output' pane on the right shows the same columns, with 'ProductCategory' highlighted. The 'Properties' pane on the far right shows the 'Name' property set to 'ProductCategory'.

- i. We need to define a central type. We can do this by selecting the **ProductId** column in the Output and setting Key Attribute to True.

The screenshot shows the 'Output' and 'Properties' panels. The 'Output' panel lists columns: ProductId, TypeCode, ProductCategory, Currency, Price, ProductName, and ProductDesc. The 'Properties' panel shows the 'Key Attribute' property set to 'True'.

Property	Value
Name	ProductId
Label	ProductId
Mapping	SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data::products.f
Key Attribute	True
Drill Down Enabled	False
Label Column	

- j. Drag and drop the **NameId** column from the **products** table to the **TextId** column of the **texts** table. Change the Join Type to **Text Join** and select **Language** as the Language Column.

The screenshot shows the SAP HANA Studio interface with three tables: **products**, **addresses**, and **businessPartner**. The **products** table has columns like **ProductId**, **TypeCode**, **Category**, **CreatedBy**, **CreatedAt**, **ChangedBy**, **ChangedAt**, **NameId**, **DescId**, **SupplierId**, **TaxTariffCode**, **QuantityUnit**, **WeightMeasure**, **WeightUnit**, **Currency**, **Price**, **ProductPicURL**, **Width**, **Depth**, **Height**, and **DimensionUnit**. The **texts** table has columns like **TextId**, **Language**, **ISOLanguage**, and **Text**. The **addresses** table has columns like **AddressId**, **City**, **PostalCode**, **Street**, **Building**, **Country**, **AddressType**, **StartDate**, and **EndDate**. The **businessPartner** table has columns like **PartnerId**, **PartnerRole**, **EmailAddress**, **PhoneNumber**, **FaxNumber**, **WebAddress**, **AddressId**, **CompanyName**, **LegalForm**, **CreatedBy**, **CreatedAt**, **ChangedBy**, **ChangedAt**, and **Currency**. The **Join Type** is set to **Text Join** and the **Language Column** is set to **Language**.

- k. Add **Text** from texts table to the output and change the column name to **ProductName**.

The screenshot shows the SAP HANA Studio interface with the same three tables as before. The **Text** column from the **texts** table is being added to the output. The column name is changed to **ProductName**. The **Join Type** is set to **Text Join** and the **Language Column** is set to **Language**.

- I. Create an Inner, 1:n join between **SupplierId** of the **product** table and the **PartnerId** column of the **businessPartner**. Add **PartnerId** and **CompanyName** to the output from **businessPartner** table.

Details

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:products

- ProductID: Product ID
- TypeCode: Product Type Code
- Category: Product Category
- CreatedBy: Created By
- CreatedAt: Created At - Date an...
- ChangedBy: Last Changed By
- ChangedAt: Last Changed At - D...
- Named: Product Name ID
- DescId: Product Description ID
- SupplierId: Supplier ID
- TaxTariffCode: Tax Tariff Code
- QuantityUnit: Quantity Unit
- WeightMeasure: Weight Measure
- WeightUnit: Weight Unit
- Currency: Currency Code
- Price: Product Unit Price
- ProductPicURL: URL of Product Image
- Width: Width Dimension
- Depth: Depth Dimension
- Height: Height Dimension
- DimensionUnit: Dimension Unit

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:addresses

- AddressId: Address ID
- City: City
- PostalCode: Postal Code
- Street: Street
- Building: Building
- Country: Country
- AddressType: Address Type
- StartDate: Validity Start Date
- EndDate: Validity End Date

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:businessPartner

- PartnerId: Partner ID
- PartnerRole: Business Partner Role
- EmailAddress: E-Mail Address
- PhoneNumber: Business Phone Nu...
- FaxNumber: Business Fax Number
- WebAddress: Web Address
- AddressId: Address ID
- CompanyName: Company Name
- LegalForm: Legal Form of Busine...
- CreatedBy: Created By
- CreatedAt: Created At - Date an...
- ChangedBy: Last Changed By
- ChangedAt: Last Changed At - D...
- Currency: Currency Code

Output

Columns

- ProductId: sap.hana.democontent.epm.data:
- TypeCode: sap.hana.democontent.epm.data:
- ProductCategory: sap.hana.democontent.ep
- Currency: sap.hana.democontent.epm.data:
- Price: sap.hana.democontent.epm.data:prod
- ProductName: sap.hana.democontent.epm.c
- PartnerId: sap.hana.democontent.epm.data:
- CompanyName: sap.hana.democontent.epm
- Calculated Columns

Properties

Property	Value
Left Element	SAP_HANA_EPM_DEMO.sa
Right Element	SAP_HANA_EPM_DEMO.sa
Join Type	Inner
Language Column	
Cardinality	1..n

- m. Add an Inner, 1:1 join between the **AddressId** of the **businessPartner** table to the **AddressId** of the **addresses** table. Add the **City**, **PostalCode**, **Street**, **Building**, and **Country** columns of **addresses** to the output.

Details

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:products

- ProductID: Product ID
- TypeCode: Product Type Code
- Category: Product Category
- CreatedBy: Created By
- CreatedAt: Created At - Date an...
- ChangedBy: Last Changed By
- ChangedAt: Last Changed At - D...
- Named: Product Name ID
- DescId: Product Description ID
- SupplierId: Supplier ID
- TaxTariffCode: Tax Tariff Code
- QuantityUnit: Quantity Unit
- WeightMeasure: Weight Measure
- WeightUnit: Weight Unit
- Currency: Currency Code
- Price: Product Unit Price
- ProductPicURL: URL of Product Image
- Width: Width Dimension
- Depth: Depth Dimension
- Height: Height Dimension
- DimensionUnit: Dimension Unit

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:addresses

- AddressId: Address ID
- City: City
- PostalCode: Postal Code
- Street: Street
- Building: Building
- Country: Country
- AddressType: Address Type
- StartDate: Validity Start Date
- EndDate: Validity End Date

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:businessPartner

- PartnerId: Partner ID
- PartnerRole: Business Partner Role
- EmailAddress: E-Mail Address
- PhoneNumber: Business Phone Nu...
- FaxNumber: Business Fax Number
- WebAddress: Web Address
- AddressId: Address ID
- CompanyName: Company Name
- LegalForm: Legal Form of Busine...
- CreatedBy: Created By
- CreatedAt: Created At - Date an...
- ChangedBy: Last Changed By
- ChangedAt: Last Changed At - D...
- Currency: Currency Code

Output

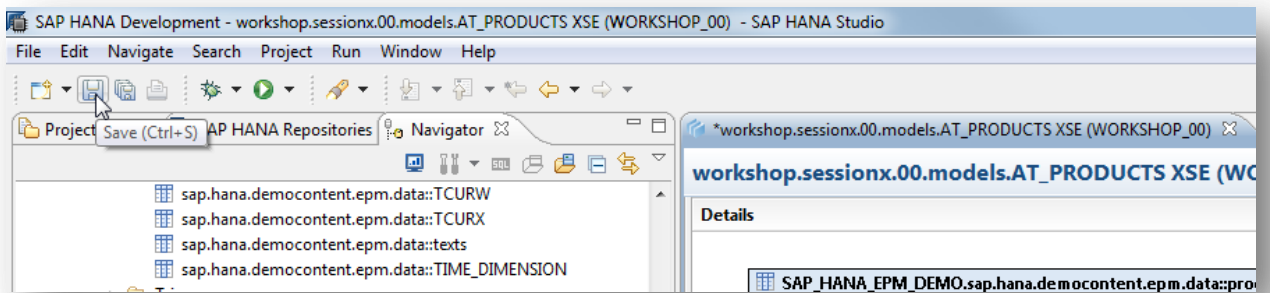
Columns

- ProductId: sap.hana.democontent.epm.data:
- TypeCode: sap.hana.democontent.epm.data:
- ProductCategory: sap.hana.democontent.ep
- Currency: sap.hana.democontent.epm.data:
- Price: sap.hana.democontent.epm.data:prod
- ProductName: sap.hana.democontent.epm.c
- PartnerId: sap.hana.democontent.epm.data:
- CompanyName: sap.hana.democontent.epm
- City: sap.hana.democontent.epm.data:adre
- PostalCode: sap.hana.democontent.epm.dat
- Street: sap.hana.democontent.epm.data:ad
- Building: sap.hana.democontent.epm.data:a
- Country: sap.hana.democontent.epm.data:a
- Calculated Columns

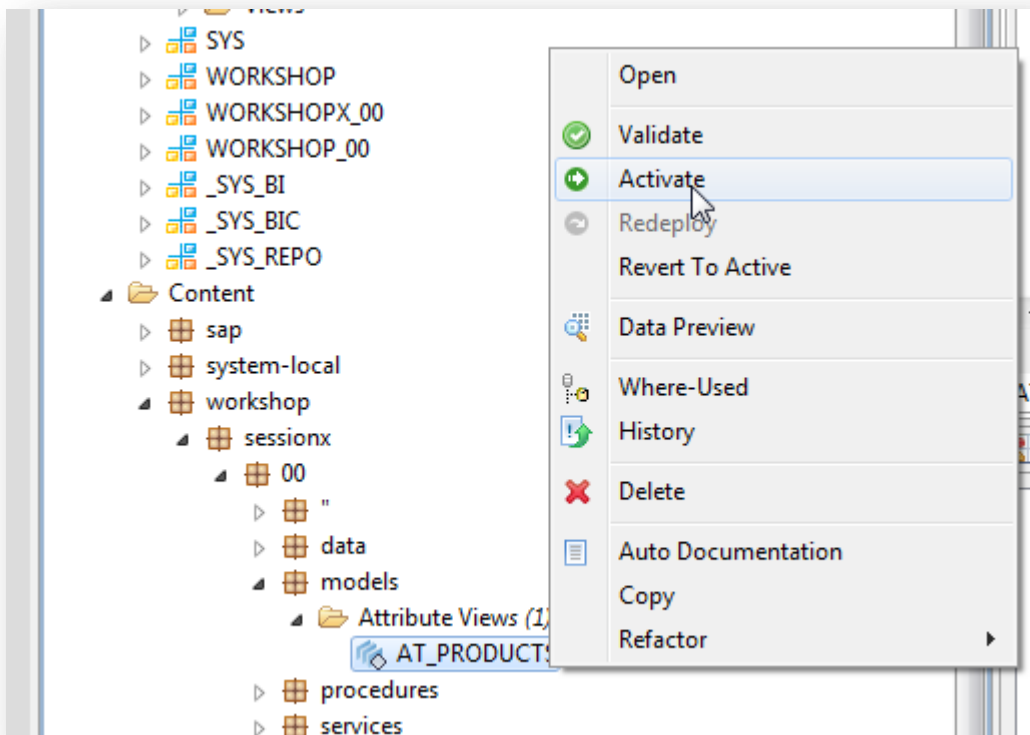
Properties

Property	Value
Left Element	SAP_HANA_EPM_DEMO.sa
Right Element	SAP_HANA_EPM_DEMO.sa
Join Type	Inner
Language Column	
Cardinality	1..1

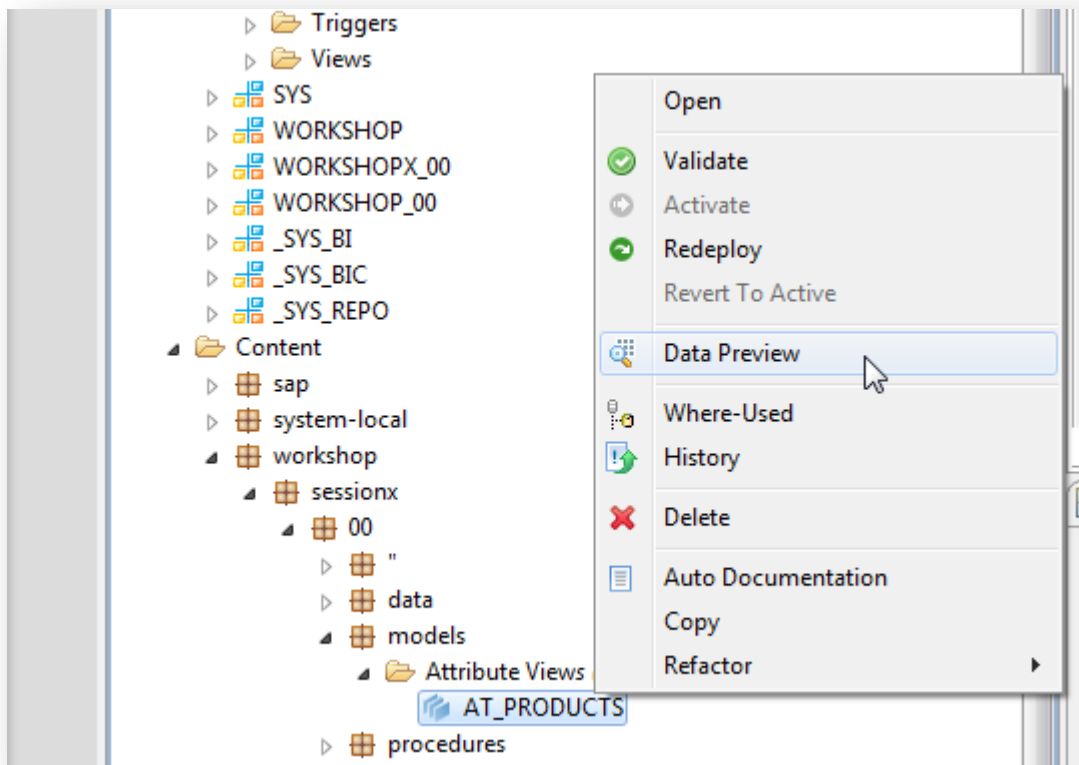
n. Save your model



o. Activate your model



- p. To test your Attribute View, choose **Data Preview** from the context menu.

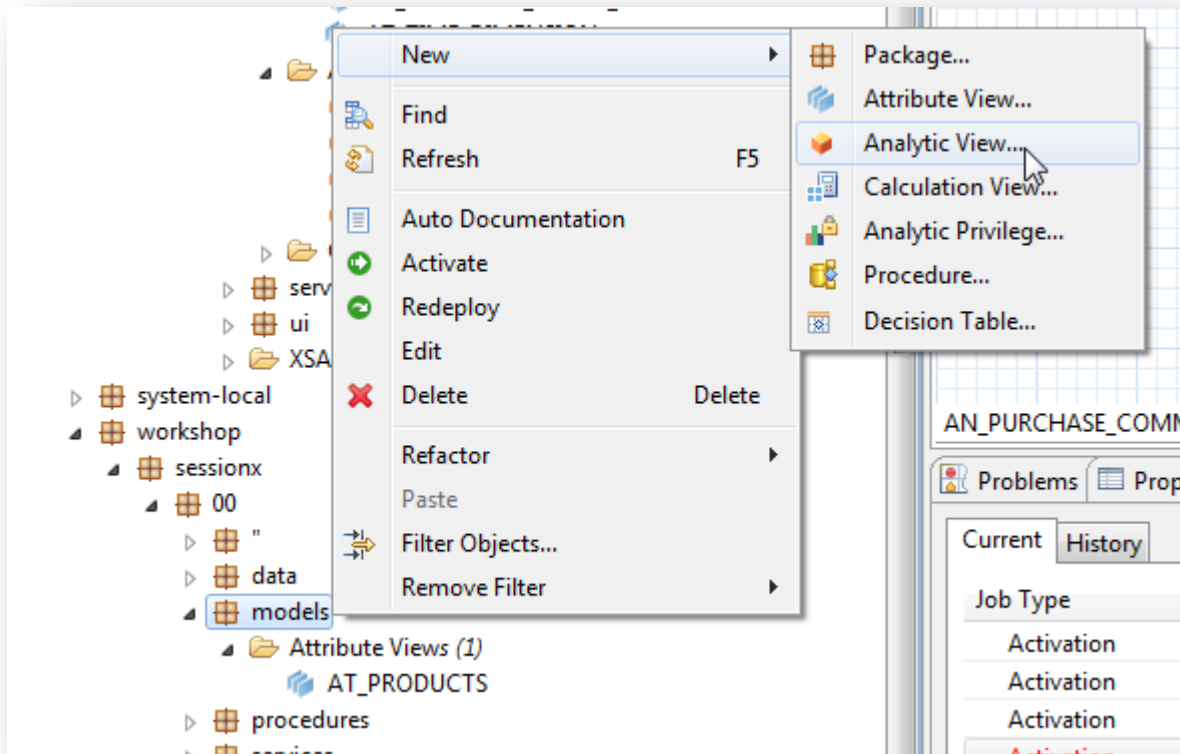


- q. In the data preview, there are many tools for performing analysis on the view data. For an initial test choose Raw Data and make sure your outlook looks similar to the following:

Analysis											Distinct values											Raw Data											Show Log											Max rows: 200										
Filter pattern											106 rows retrieved - 213 ms											Add filter																																
RB	TypeCode	RB	ProductCategory	RB	Currency	12	Price	RB	ProductName	RB	PartnerId	RB	CompanyName	RB	City	RB	PostalCode	RB	Street	RB	Building	RB	Country	RB	ProductId																													
PR			Notebooks		EUR		956		Notebook Basic 15		0100000000		SAP		Walldorf		69190		Dietmar-H...		15		DE		HT-1000																													
PR			Notebooks		EUR		1,249		Notebook Basic 17		0100000001		Becker Berlin		Berlin		13467		Calvinstrasse		36		DE		HT-1001																													
PR			Notebooks		USD		1,570		Notebook Basic 18		0100000002		DelBont Industries		Wilmington...		19899		1 2345 King...		01		US		HT-1002																													
PR			Notebooks		EUR		1,650		Notebook Basic 19		0100000003		Talpa		Hannover		30625		An der Bret...		422		DE		HT-1003																													
PR			Handhelds		USD		499		UMTS PDA		0100000004		Panorama Studios		Hollywoo...		91602		Morning Dr...		178		US		HT-1007																													
PR			Notebooks		EUR		1,999		Notebook Professio...		0100000005		TECUM		Muenster		48155		Albersloher...		24		DE		HT-1010																													
PR			Notebooks		JPY		2,299		Notebook Professio...		0100000006		Asia High tech		Tokyo		1000004		1-7-2 Ohte...		1		JP		HT-1011																													
PR			Handhelds		EUR		129		Easy Hand III		0100000007		Laurent		Paris		75015		Rue de la R...		1		FR		HT-1020																													
PR			Handhelds		MXN		149		Easy Hand V		0100000008		AVANTEL		Mexiko City		17000		Bosque de ...		24		MX		HT-1021																													
PR			Handhelds		ARS		205		Easy Hand VII		0100000009		Telecomunicaciones ...		Buenos Ai...		1511		Av. Liberta...		12221		AR		HT-1022																													
PR			Handhelds		USD		239		Easy Power		0100000010		Pear Computing Servi...		Atlanta G...		30338		Dunwoody ...		3000		US		HT-1023																													
PR			Flat screens		EUR		230		Ergo Screen		0100000011		Alpine Systems		Salzburg		5010		Arnold Sch...		54		AT		HT-1030																													
PR			Flat screens		GBP		285		Easy Pivel		0100000012		New Line Design		Manchester		M4 8RE		Kelley Street		97		GB		HT-1031																													
PR			Flat screens		EUR		345		Beam Screen		0100000013		HEPA Tec		Bremen		28213		Theodor H...		43		DE		HT-1032																													
PR			Flat screens		USD		399		Flat Basic 15		0100000014		Anav Ideon		Bismarck ...		58501		Franklin Ave		153		US		HT-1035																													
PR			Flat screens		CAD		430		Flat Future		0100000015		Robert Brown Entertai...		Quebec		JUL 1T0		Rue Sault A...		65		CA		HT-1036																													
PR			Flat screens		MXN		1,430		Flat X-large		0100000016		Mexican Oil Trading ...		Puebla		40615		Av. Herma...		366		MX		HT-1037																													
PR			Laser printers		EUR		830		Laser Professional E...		0100000017		Meliva		Kln		50827		Robert Koc...		14		DE		HT-1040																													
PR			Laser printers		ARS		490		Laser Basic		0100000018		Compostela		Mendoza		M5500		Avenida De...		53		AR		HT-1041																													
PR			Laser printers		EUR		349		Laser Allround		0100000019		Pateu		Lyon		69007		Quai de Pie...		81		FR		HT-1042																													

2. Analytic View

- a. Now we will create an Analytic View which combines purchase order table data with the product attribute view we created in the previous step.
- b. In your models package, create a new Analytic View



- c. Name your new view **AN_PURCHASE_ORDERS**

Create an Information View
Select the required view type and enter the details

Name:* AN_PURCHASE_ORDERS

Description: EPM Purchase Orders

Package:* workshop.sessionx.00.models Browse...

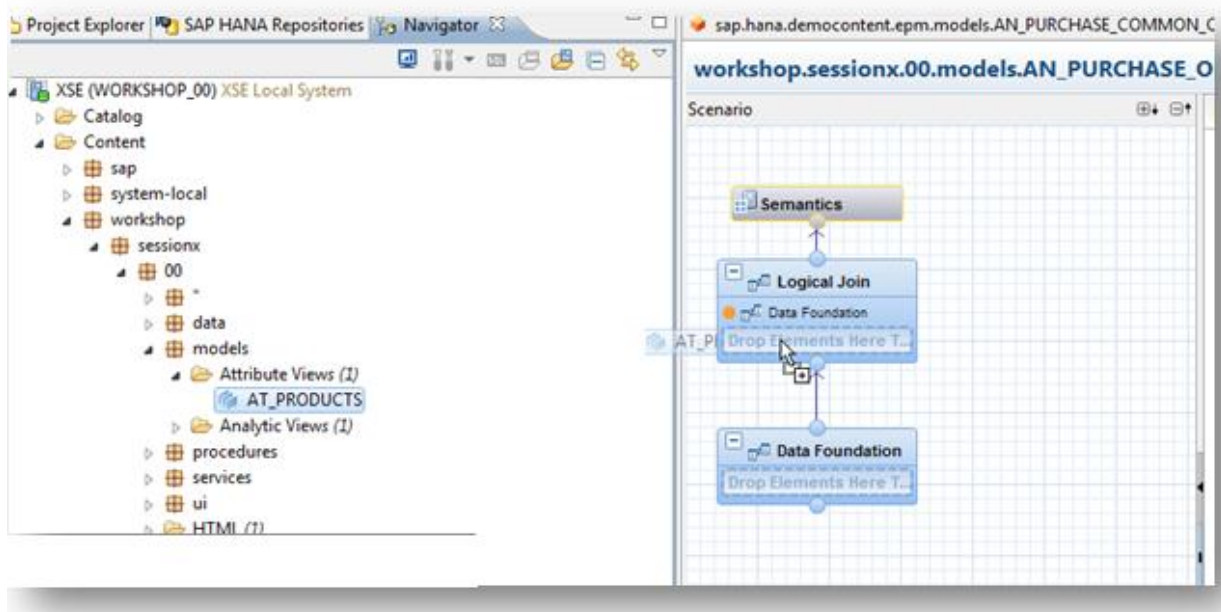
View Type: Analytic View

☐ Copy From: Browse...

Subtype:

? Finish Cancel

- d. Drag and drop the **AT_PRODUCTS** attribute view from the previous part of the exercise into the **logical Join** of your new view.



- e. Add the **purchaseOrder** and **purchaseOrderItem** tables to the Data Foundation of your view.

workshop.sessionx.00.models.AN_PURCHASE_ORDERS XSE (WORKSHOP_00)

Scenario

Semantics

Logical Join

Data Foundation

AT_PRODUCTS

Drop Elements Here T...

Data Foundation

sap.hana.democontent

sap.hana.democontent

Drop Elements Here T...

Details

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrder

PurchaseOrderId	Purchase Order ID
CreatedBy	Created By
CreatedAt	Created At - Date an...
ChangedBy	Last Changed By
ChangedAt	Last Changed At - D...
Noteld	PO Note Text ID
PartnerId	Partner ID
Currency	Currency Code
GrossAmount	Total Gross Amount
NetAmount	Total Net Amount
TaxAmount	Total Tax Amount
LifecycleStatus	PO Lifecycle Status
ApprovalStatus	PO Approval Status
ConfirmStatus	PO Confirmation Sta...
OrderingStatus	PO Ordering Status
InvoicingStatus	PO Invoicing Status

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrderItem

PurchaseOrderId	Purchase Order ID
PurchaseOrderItem	Purchase Order Item
ProductId	Product ID
Noteld	PO Note Text ID
Currency	Currency Code
GrossAmount	Gross Amount
NetAmount	Net Amount
TaxAmount	Tax Amount
Quantity	Quantity
QuantityUnit	Quantity Unit
DeliveryDate	Scheduled Delivery D...

- f. Create a **1:n referential** join between **purchaseOrder** and **purchaseOrderItem** on the **PurchaseOrderId** column.

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrder

PurchaseOrderId	Purchase Order ID
CreatedBy	Created By
CreatedAt	Created At - Date an...
ChangedBy	Last Changed By
ChangedAt	Last Changed At - D...
Noteld	PO Note Text ID
PartnerId	Partner ID
Currency	Currency Code
GrossAmount	Total Gross Amount
NetAmount	Total Net Amount
TaxAmount	Total Tax Amount
LifecycleStatus	PO Lifecycle Status
ApprovalStatus	PO Approval Status
ConfirmStatus	PO Confirmation Sta...
OrderingStatus	PO Ordering Status
InvoicingStatus	PO Invoicing Status

SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrderItem

PurchaseOrderId	Purchase Order ID
PurchaseOrderItem	Purchase Order Item
ProductId	Product ID
Noteld	PO Note Text ID
Currency	Currency Code
GrossAmount	Gross Amount
NetAmount	Net Amount
TaxAmount	Tax Amount
Quantity	Quantity
QuantityUnit	Quantity Unit
DeliveryDate	Scheduled Delivery D...

Columns

Input Parameters

Properties

General

Property	Value
Left Element	SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrder
Right Element	SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data:purchaseOrderItem
Join Type	Referential
Language Column	
Cardinality	1..n

- g. Using Add to Output, add the column **CreatedAt** from the purchaseOrder table and the **PurchaseOrderId**, **PurchaseOrderItem**, **ProductId**, **Currency**, and **GrossAmount** columns from the purchaseOrderItem table.

Table	Column	Description
SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data::purchaseOrder	PurchaseOrderId	Purchase Order ID
	CreatedBy	Created By
	CreatedAt	Created At - Date a...
	ChangedBy	Last Changed By
	ChangedAt	Last Changed At - D...
	NoteId	PO Note Text ID
	PartnerId	Partner ID
	Currency	Currency Code
	GrossAmount	Total Gross Amount
	NetAmount	Total Net Amount
	TaxAmount	Total Tax Amount
	LifecycleStatus	PO Lifecycle Status
	ApprovalStatus	PO Approval Status
	ConfirmStatus	PO Confirmation Sta...
	OrderingStatus	PO Ordering Status
	InvoicingStatus	PO Invoicing Status
	SAP_HANA_EPM_DEMO.sap.hana.democontent.epm.data::purchaseOrderItem	PurchaseOrderId
PurchaseOrderItem		Purchase Order Item
ProductId		Product ID
NoteId		PO Note Text ID
Currency		Currency Code
GrossAmount		Gross Amount
NetAmount		Net Amount
TaxAmount		Tax Amount
Quantity		Quantity
QuantityUnit		Quantity Unit
DeliveryDate	Scheduled Delivery D...	

- h. From the **Semantics Scenario**, Set the Default Schema to **SAP_HANA_EPM_DEMO**:

workshop.sessionx.00.models.AN_PURCHASE_ORDERS XSE (WORKSHOP_00)

Scenario: AN_PURCHASE_ORDERS XSE (WORKSHOP_00)

Details:

Properties:

- AN_PURCHASE_ORDERS XSE.workshop.sessionx.00.models
- EPM Purchase Orders
- Multi Dimensional Reporting: True
- Default Schema: SAP_HANA_EPM_DEMO
- Default Client: Dynamic

- i. Select the Logical Join box in the Scenario. You can then drag and drop the **ProductId_1** from the data foundation to the **ProductId** column of the AT_PRODUCTS view creating a 1:1 referential join.

workshop.sessionx.00.models.AN_PURCHASE_ORDERS XSE (WORKSHOP_00)

Scenario

Details

workshop.sessionx.00.models.AT_PRODUCTS

Column	Type
ProductId	ProductId
TypeId	TypeId
ProductCategory	Product Category
Currency	Currency
Price	Price
ProductName	Product Name
ProductDesc	Product Description
PartnerId	PartnerId
CompanyName	Company Name
City	City
PostalCode	Postal Code
Street	Street
Building	Building
Country	Country

Data Foundation

Column	Type
CreatedAt	CreatedAt
PurchaseOrderId	PurchaseOrderItem
PurchaseOrderItem	PurchaseOrderItem
ProductId_1	ProductId_1
Currency_1	Currency_1
GrossAmount	GrossAmount

Output

Properties

Property	Value
Left Element	Data Foundation.ProductId_1
Right Element	workshop.sessionx.00.models...
Join Type	Referential
Cardinality	1..1
Temporal Column	
From Column	
To Column	
Temporal Condition	

- j. Return to the Semantics Scenario, set the **GrossAmount** as measures and the other fields as attributes.

Scenario

Details

Properties

AN_PURCHASE_ORDERS

HDB:workshop.sessionx.01.models

AN_PURCHASE_ORDERS

Column

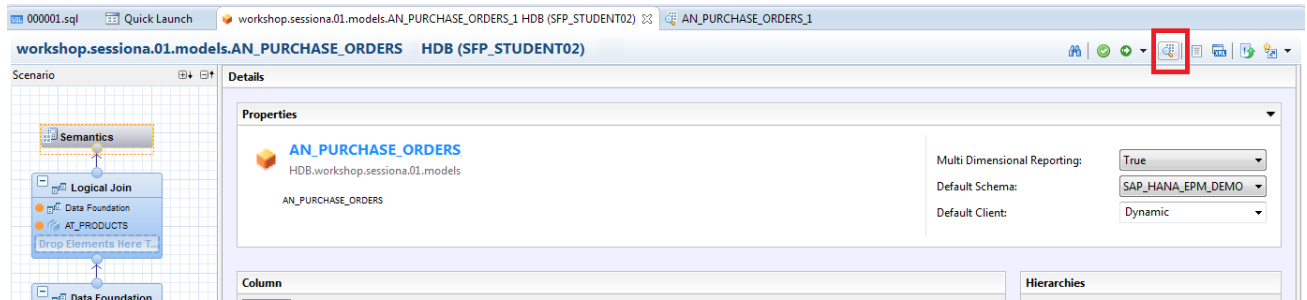
Type	Name	Label	Aggregation	Variable	Label Column	Hidden
	CreatedAt	CreatedAt				<input type="checkbox"/>
AB	PurchaseOrderId	PurchaseOrder...				<input type="checkbox"/>
AB	PurchaseOrderItem	PurchaseOrder...				<input type="checkbox"/>
AB	Currency_1	Currency_1				<input type="checkbox"/>
12	GrossAmount	GrossAmount	SUM			<input type="checkbox"/>

Hierarchies

Name	Label	Source

Variables/Input Parameters

- k. Save and activate your Analytic View and then do data preview. (You can either right click on view name and select data preview or simply click on data preview icon)

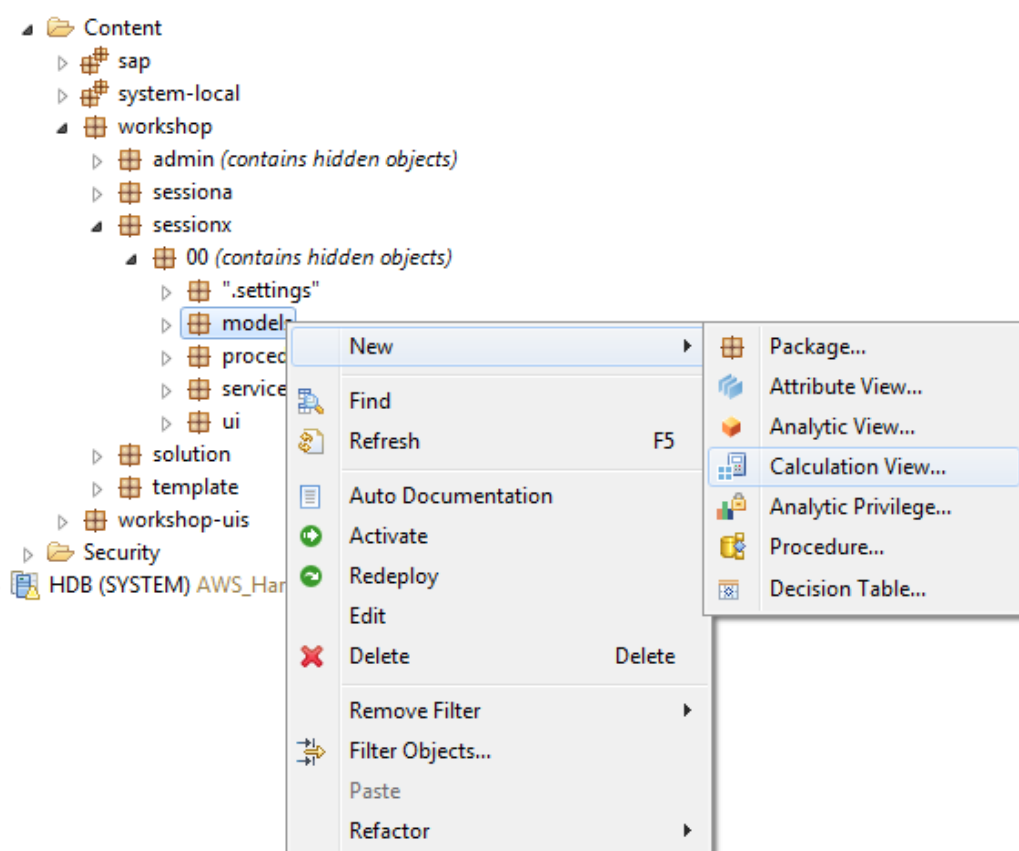


- l. You should see output similar to this.

TypeCode	ProductCategory	Currency	Price	ProductName	ProductDescription	PartnerId	CompanyName	ProductId	CreatedAt	PurchaseOrderId
PR	Flat-screen television	ARS	1,199	Flat Watch HD37	1000000146	0100000027	Developement Para O...	HT-6131	Jul 2, 2012	0300000183
PR	Notebooks	EUR	956	Notebook Basic 15	1000000002	0100000000	SAP	HT-1000	Jun 28, 2012	1300000545
PR	PC	EUR	2,399	PC Power Station	1000000136	0100000022	Quimica Madrilenos	HT-1210	Apr 2, 2012	1300000824
PR	Flat-screen television	EUR	1,459	Flat Watch HD32	1000000144	0100000026	Vente Et Réparation d...	HT-6130	Jul 11, 2012	0300000558
PR	Others	EUR	6.9	Notebook Lock	1000000114	0100000011	Alpine Systems	HT-1111	Dec 17, 2012	1300000351
PR	Graphic cards	EUR	24	Psimax	1000000078	0100000038	Bionic Research Lab	HT-1073	May 1, 2012	1300000487
PR	Electronics	USD	449.99	10 Portable DVD pla...	1000000190	0100000004	Panorama Studios	HT-2001	Feb 17, 2012	1300000779
PR	Others	ARS	79	USB Stick 1 GB	1000000128	0100000018	Compostela	HT-1118	Dec 24, 2012	1300000358
PR	Ink-jet printers	EUR	170	Deskjet Super Highs...	1000000046	0100000022	Quimica Madrilenos	HT-1052	Nov 20, 2012	1300000324
PR	Others	MXN	36	ISDN direct	1000000124	0100000016	Mexican Oil Trading ...	HT-1116	May 16, 2012	0300000136
PR	Scanner	CNY	520	Jet Scan Professional	1000000086	0100000042	Siwusha	HT-1083	Apr 9, 2012	0300000465
PR	Software	USD	69	Smart Network	1000000100	0100000004	Panorama Studios	HT-1102	May 30, 2012	0300000516
PR	Ink-jet printers	USD	99	Deskjet Mobile	1000000044	0100000021	Florida Holiday Comp...	HT-1051	Dec 6, 2012	0300000340
PR	Graphic cards	ZAR	81.7	Gladiator MX	1000000074	0100000036	African Gold And Dia...	HT-1071	Apr 29, 2012	0300000485

3. Calculation View

- a. There are several ways to create views in the SAP HANA database. We just learned about analytic and attribute views and how to combine them into a star schema which forms an analytic view itself. These views are completely processed by the column store. For more complex views we need to leverage the power of the built-in functions. These views are called calculation views. In this example we will create the example GC_TAX_RATE as a graphical calculation view.
- b. Right-click on your new package and create a new Calculation View. Note that this will also create a new folder in which to store the view.



- c. Give your new graphical calculation view a name **GC_TAX_RATE** and click finish.

New Calculation View

This wizard creates a calculation view in the specified package.

Name:* GC_TAX_RATE

Description Calculate Tax Rate

Package:* workshop.sessionx.00.models Browse...

☒ Create New

☐ Copy From Browse...

View Type

☒ Graphical

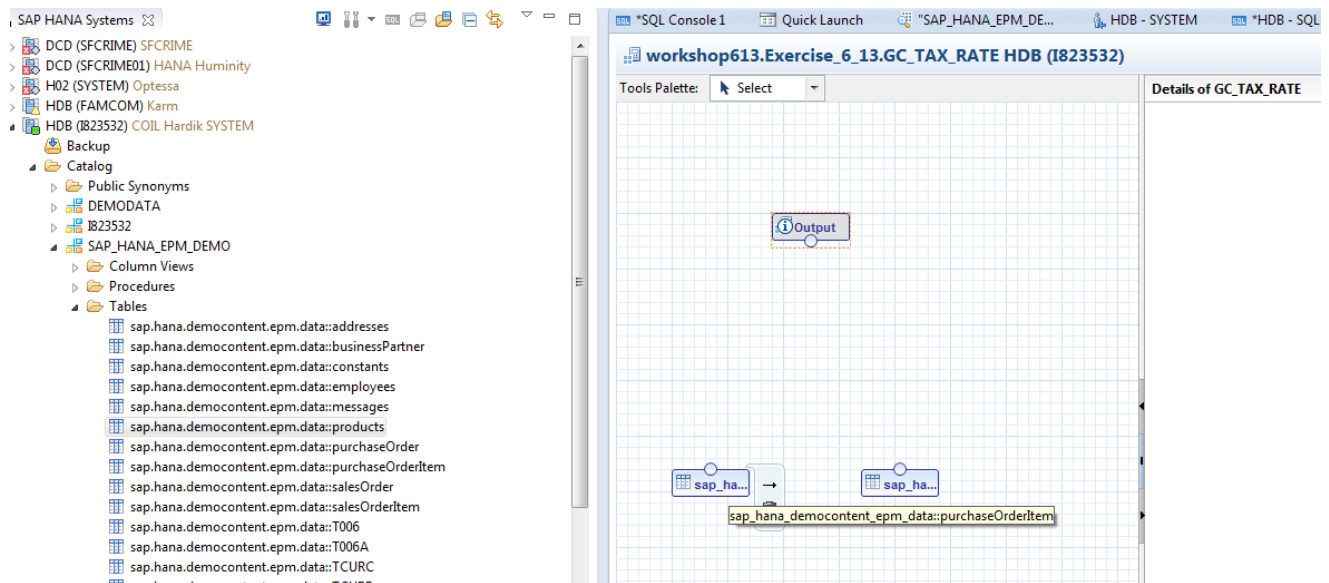
☐ SQL Script

Schema for conversion: SAP_HANA_EPM_DEMO

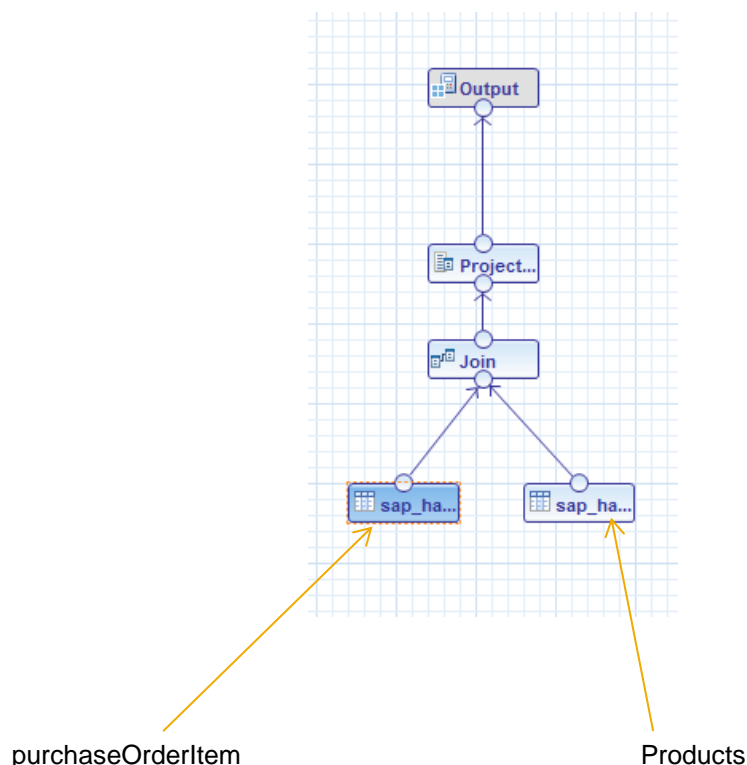
Run With Definer's Rights (User_SYS_REPO)

? < Back Next > Finish Cancel

- d. Now, drag and drop two tables from SAP_HAHA_EPM_DEMO schema in to the view. 1) **purchaseOrderItem** 2) **Products**. The modeler canvas will appear and it is split into several areas. The left one is used to connect tables with the output node by means of various operations which can be picked from the Tools Palette on top of that area.



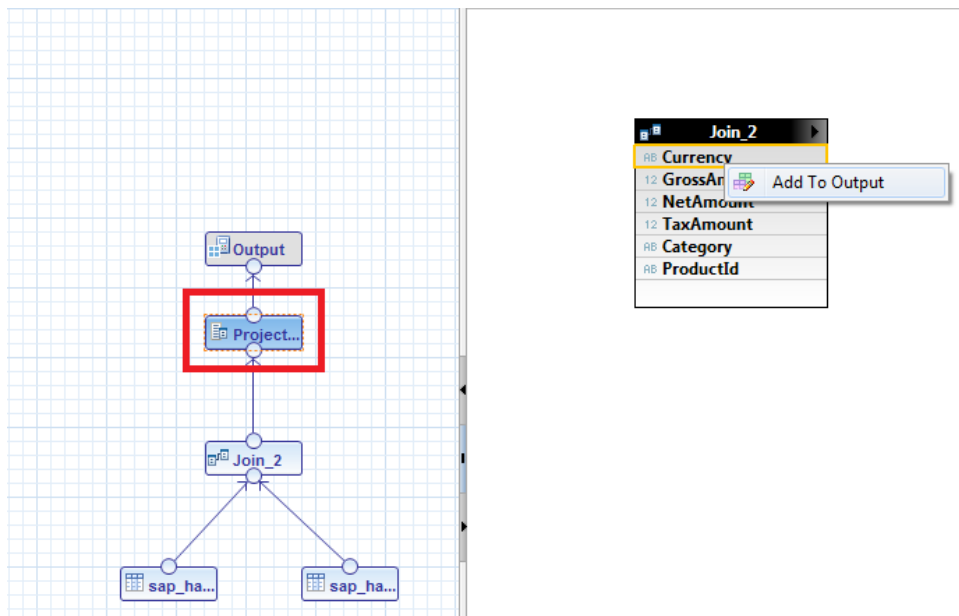
- e. Add the **Projection** and **Join** operation from the Tools Palette. Connect **purchaseOrderItem** and **products** with **Join**, Join with **Projection** and Projection with **Output**.



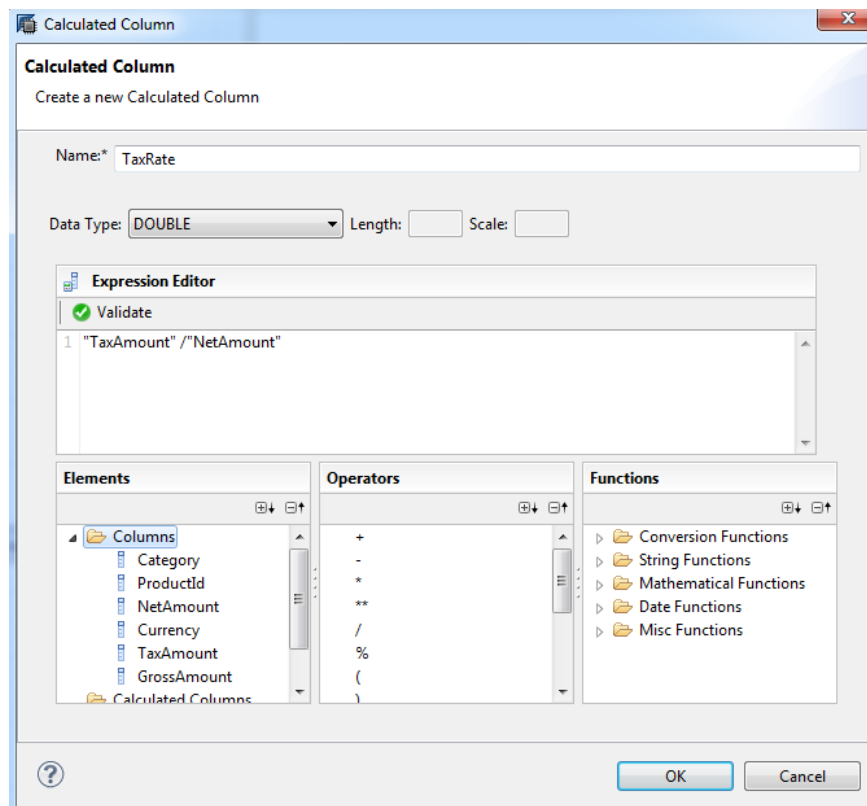
- f. Now, select Join and make 1..n inner Join for ProductId from purchaseOrderItem table to products table. Also, add **Currency**, **GrossAmount**, **NetAmount**, **TaxAmount** to output from purchaseOrderItem table and **Category** from products table.

sap_hana_democontent_epm_data::purchaseOrderItem	sap_hana_democontent_epm_data::products
AB PurchaseOrderId	AB ProductId
AB PurchaseOrderItem	AB TypeCode
AB ProductId	AB Category
AB NotedId	AB CreatedBy
AB Currency	AB CreatedAt
12 GrossAmount	AB ChangedBy
12 NetAmount	AB ChangedAt
12 TaxAmount	AB NameId
12 Quantity	AB DescId
AB QuantityUnit	AB SupplierId
AB DeliveryDate	12 TaxTariffCode
	AB QuantityUnit
	12 WeightMeasure
	AB WeightUnit
	AB Currency
	12 Price
	AB ProductPicURL
	12 Width
	12 Depth
	12 Height
	AB DimensionUnit

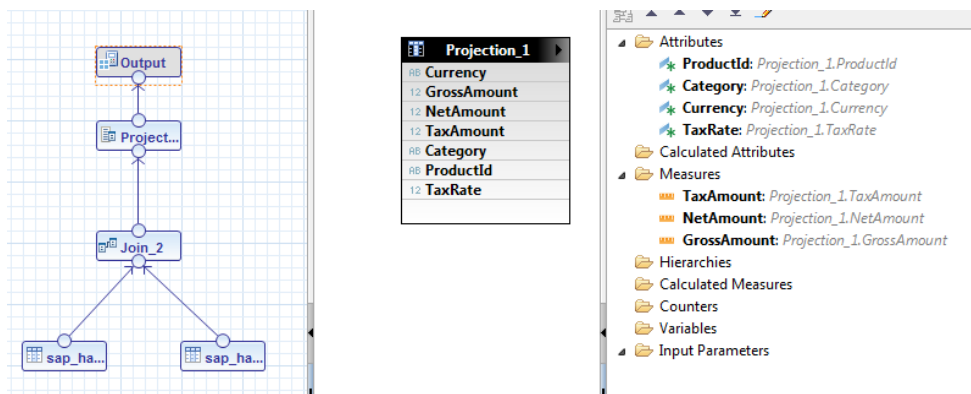
- g. Now, select Projection and add **Currency**, **GrossAmount**, **NetAmount**, **TaxAmount**, **Category** and **ProductId** as output from the details of Projection.



- h. Now, right click on Calculated Columns from **output of Projection** and select new. Name it as TaxRate. We will count the tax rate for the product price. Type "**TaxAmount**" / "**NetAmount**" in Expression Editor. You can also select this columns from Elements. At the end, Validate your expression and select "Add".



- i. Now, Select output and add **Currency, Category, ProductId** and **TaxRate** as attributes and **TaxAmount, NetAmount** and **GrossAmount** as Measures.



- j. Now, we will create Input Parameters for this view to take an input for category. Right click on Input Parameters and select new. Name it IP_CATEGORY. Make sure that you have selected output.

Edit Input Parameter

Define the properties of the Input Parameter

Name:* IP_CATEGORY

Description: IP_CATEGORY

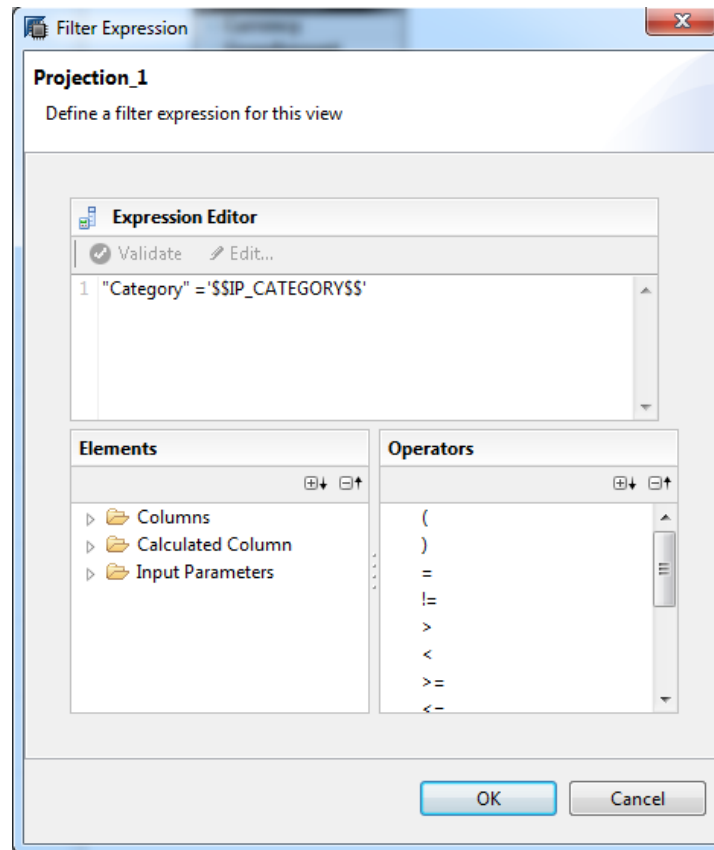
Type: Attribute Value

Default Value: ☐ Is Mandatory

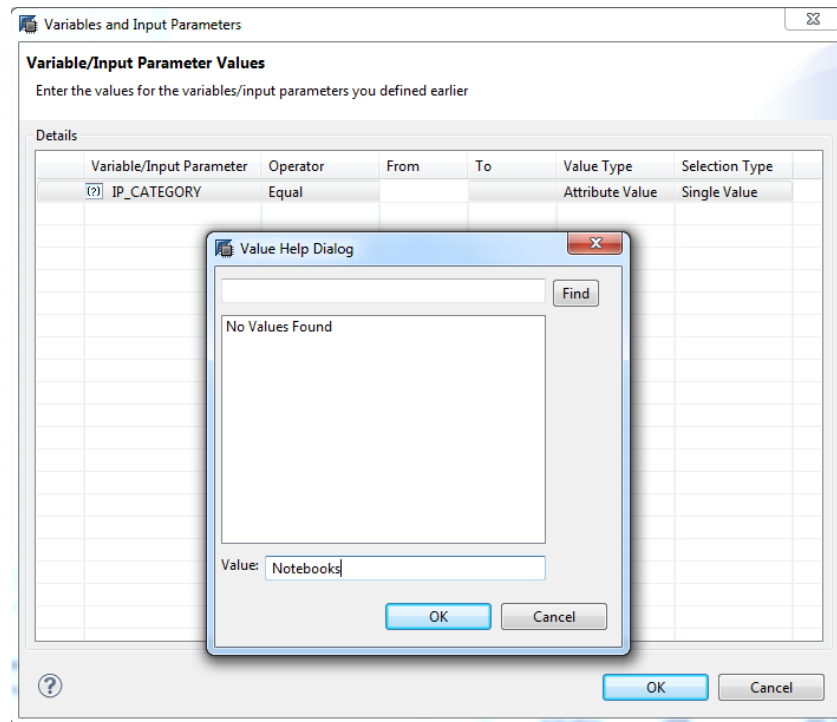
Attribute Value

Attribute:* Category

- k. Now, go back to Projection and double click on Expression. The filter expression window will open. In expression editor type "Category" = '\$\$IP_CATEGORY\$\$' and select ok. You can select columns from Elements if you want.



- I. Save and activate the view and then do data preview. In the input parameters window, you can enter either Software, Graphic cards, others, High Tech or notebooks and select ok.



- j. You should be able see the data like below.

Analysis

Distinct values

Raw Data

Filter pattern

9 rows retrieved - 31 ms

AB	Category	AB	ProductId	AB	Currency	12	TaxRate	12	GrossAmount	12	NetAmount
	Notebooks		HT-1000		EUR		0.19		209,325.76		175,904
	Notebooks		HT-1002		USD		0.19		298,928		251,200
	Notebooks		HT-1003		EUR		0.19		361,284		303,600
	Notebooks		HT-1010		EUR		0.19		589,944.88		495,752
	Notebooks		HT-1011		JPY		0.19		853,572.72		717,288
	Notebooks		HT-8000		EUR		0.19		131,211.78		110,262
	Notebooks		HT-8001		GBP		0.19		173,566.26		145,854
	Notebooks		HT-8002		EUR		0.19		368,116.98		309,342
	Notebooks		HT-8003		USD		0.19		198,206.4		166,560

© 2012 by SAP AG. All rights reserved.

SAP and the SAP logo are registered trademarks of SAP AG in Germany and other countries. Business Objects and the Business Objects logo are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company. Sybase and the Sybase logo are registered trademarks of Sybase Inc. Sybase is an SAP company. Crossgate is a registered trademark of Crossgate AG in Germany and other countries. Crossgate is an SAP company.



The Best-Run Businesses Run SAP™