



LOGALI

Teoría
CDS Analíticos

Cambiar





Contenido

5. CDS Analíticos	3
5.1. VDM – Virtual Data Model	3
5.2. Categorías de datos analíticos	6
5.3. Transient View Entity	9
5.4. Localized	12
5.5. Virtual Element	13
5.6. Analytic Query – Developer Mode	13



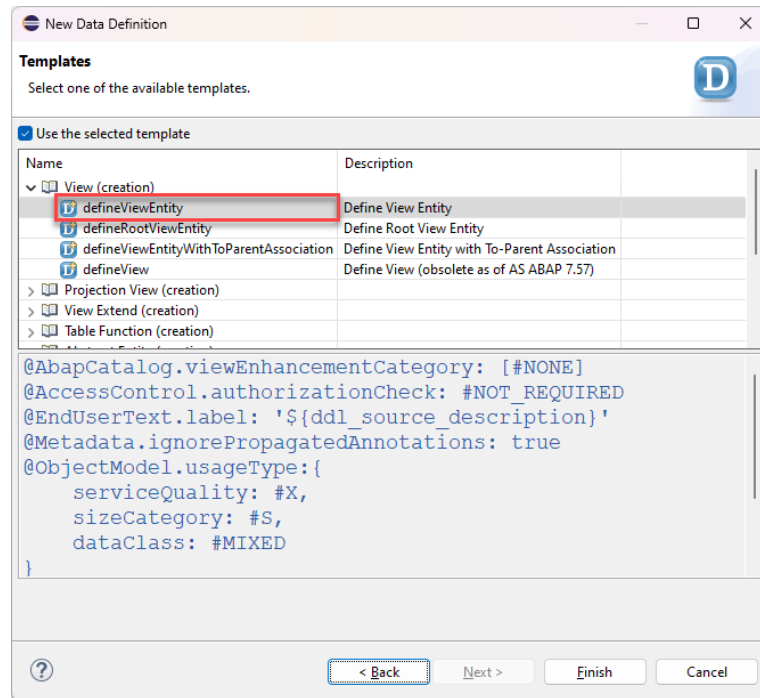
5. CDS Analíticos

5.1. VDM – Virtual Data Model

Es una herramienta esencial en SAP HANA que permite la creación de modelos virtuales organizados en diferentes capas. Estos modelos facilitan la extracción, análisis y uso de datos en diversas aplicaciones analíticas. Utilizando los Core Data Services (CDS), el VDM habilita una estructura modular y eficiente para gestionar los datos, proporcionando capacidades avanzadas de búsqueda y **virtualización** en múltiples niveles. Esto permite a las empresas tomar decisiones basadas en datos en tiempo real, optimizando la creación de dashboards y reportes analíticos cruciales para la comprensión y gestión de la situación empresarial.

La virtualización se refiere a la creación de modelos virtuales en diferentes capas utilizando los CDS. Estos modelos pueden ser utilizados en varias categorías de datos analíticos para definir y mostrar información que puede ser interpretada por herramientas analíticas. El uso de CDS permite tener una estructura organizada y modular para la extracción y procesamiento de datos.

Para crear una entidad o vista CDS necesaria para la asociación parent child, se realiza el mismo procedimiento para crear una entidad o vista CDS a través de la carpeta de proyecto luego New en la opción **Other ABAP Repository Object** ubicar la carpeta **Core Data Services** y luego seleccionar **Data Definition** y seleccionar la plantilla **defineViewEntity** que se encuentra en la carpeta **View (creation)**.



Herramientas de Búsqueda:

El VDM también incluye capacidades de búsqueda avanzadas, como la propiedad **Searchable**, que permite configurar criterios de búsqueda específicos para los datos. Esto incluye la configuración de un umbral de tolerancia al error, que determina la precisión de los resultados de búsqueda. Un umbral más alto permite búsquedas más precisas, mientras que un umbral más bajo permite búsquedas más amplias y menos exactas. Algunas de ellas se presentan a continuación

- **@Search.searchable: true:** habilita la capacidad de búsqueda para ese campo específico dentro de la base de datos. Los usuarios podrán buscar valores dentro de este campo en los resultados de búsqueda. Se declara al inicio de la vista CDS.
- **@Search.defaultSearchElement: true:** Esta anotación establece el campo como el elemento de búsqueda predeterminado. Cuando se realiza una búsqueda sin especificar un campo, este será el utilizado por defecto. Ayuda a simplificar las consultas y asegura que las búsquedas se dirijan al campo más relevante.
- **@Search.ranking: #HIGH:** Esta anotación asigna una alta importancia al campo en los resultados de búsqueda. Los



campos con un ranking alto aparecerán más arriba en los resultados de búsqueda, asegurando que los datos más importantes o relevantes sean presentados primero. El ranking puede tener diferentes niveles, como bajo (low), medio (medium) y alto (high).

- **@Search.fuzzinessThreshold: decimal_number:** Esta anotación configura el umbral de tolerancia para la búsqueda difusa (fuzzy search). Por ejemplo el valor 0.8 significa que la búsqueda permitirá cierto grado de imprecisión, encontrando coincidencias que son aproximadamente un 80% similares al término de búsqueda. Esto es útil para encontrar resultados que pueden tener ligeras variaciones o errores tipográficos.

Capas de Virtualización: El proceso de virtualización en SAP HANA incluye varias capas y declaran al inicio de la vista CDS:

- **Capa Básica:** Relaciona las tablas y datos fundamentales. Y se establece por medio de la anotación **@VDM.viewType:#BASIC**.
- **Capa de Composición:** Modela la información a un nivel intermedio. Y se establece por medio de la anotación **@VDM.viewType:#COMPOSITE**.
- **Capa de Consumo:** Proporciona la información para su uso final en herramientas analíticas. Y se establece por medio de la anotación **@VDM.viewType:#CONSUMPTION**.

Adicionalmente, SAP incluye capas de derivación de funciones, extensiones y virtualización transaccional para proporcionar mayor flexibilidad en la creación de modelos analíticos. Las cuáles serán la siguientes a continuación:

- **@VDM.viewType:#DERIVATION_FUNCTION.**
- **@VDM.viewType:#EXTENSION.**
- **@VDM.viewType:#TRANSACTIONAL.**

**Sintaxis:**

```
@AbapCatalog.viewEnhancementCategory: [#NONE]
@AccessControl.authorizationCheck: #NOT_REQUIRED
@EndUserText.label: 'CDS Virtual Data Model'
@Metadata.ignorePropagatedAnnotations: true
@ObjectModel.usageType:{
    serviceQuality: #X,
    sizeCategory: #S,
    dataClass: #MIXED
}
@Search.searchable: true
@VDM.viewType:#BASIC
define view entity entity_name
as select from data_source_name
association [0..*] to data_source_name as _AliasName on
_AliasName.AliasComponentName =
$projection.AliasComponentName
{
    @Search.defaultSearchElement: true
    @Search.ranking: #HIGH
    @Search.fuzzinessThreshold: 0.8
    component
        ....
    /* Associations */
    _AliasName
}
```

5.2. Categorías de datos analíticos

La creación de categorías de datos analíticos en SAP HANA es un proceso que utiliza el Virtual Data Model (VDM) y las anotaciones de Core Data Services (CDS) para definir, organizar y manipular datos de manera eficiente. Las categorías de datos, como los cubos (cubes) y las dimensiones, se utilizan para agrupar y agregar datos de múltiples fuentes, facilitando su uso en herramientas analíticas. Las anotaciones analíticas permiten configuraciones avanzadas de búsqueda y agregación, optimizando la representación de datos en dashboards y reportes. En resumen, el VDM en SAP HANA es esencial



para la toma de decisiones basadas en datos en tiempo real, proporcionando una ventaja competitiva a las empresas.

Tipos de categorías de datos

@Analytics.dataCategory: #AGGREGATIONLEVEL: Define la vista como un nivel de agregación. Es una proyección de la vista del cubo que agrega datos a partir de múltiples fuentes, proporcionando resúmenes y totales necesarios para análisis avanzados.

@Analytics.dataCategory: #CUBE: Define la vista como un cubo de datos. Los cubos contienen datos factuales utilizados para consultas, permitiendo análisis multidimensionales en aplicaciones de inteligencia empresarial

@Analytics.dataCategory: #DIMENSION: Define la vista como una dimensión. Las dimensiones contienen datos maestros que proporcionan el contexto para los datos de hechos en un modelo de datos. Son esenciales para analizar y agrupar los datos en aplicaciones analíticas.

@Analytics.dataCategory: #DOCSTORE: Define la vista como un almacén de documentos. Se utiliza para manejar datos semi-estructurados o no estructurados almacenados en formato de documento, como JSON o XML, que no se ajustan a un esquema tabular rígido.

@Analytics.dataCategory: #FACT: Define la vista como un hecho. Los hechos son datos transaccionales que forman el centro del esquema estrella y se representan sin redundancia. Estos datos cuantitativos se analizan en combinación con dimensiones.

Tipos de agregaciones en los componentes de la vista CDS con una categoría de datos:

La anotación **@DefaultAggregation** se utiliza en SAP HANA para especificar cómo se deben declarar las agregaciones de los valores de un campo específico de manera predeterminada. Esto es útil en el contexto de vistas CDS (Core Data Services), donde se definen



modelos de datos virtuales que serán utilizados por herramientas analíticas. A continuación se explican los diversos tipos de agregaciones:

- **@DefaultAggregation: #AVG:** calcula el promedio de los valores en un campo específico. Es útil para obtener el valor medio de un conjunto de datos numéricos, permitiendo entender tendencias generales en los datos.
- **@DefaultAggregation: #COUNT:** cuenta el número total de valores en un campo específico. Sirve para determinar la cantidad de entradas o registros en un conjunto de datos, proporcionando una medida básica de la cantidad de datos.
- **@DefaultAggregation: #COUNT_DISTINCT:** cuenta el número de valores únicos en un campo específico, eliminando duplicados. Es ideal para identificar la cantidad de entradas únicas en un conjunto de datos, lo que es crucial para análisis que requieren datos sin repeticiones.
- **@DefaultAggregation: #FORMULA:** permite definir una fórmula personalizada para la agregación de datos. Es útil para realizar cálculos complejos o específicos que no pueden ser abordados con funciones de agregación estándar.
- **@DefaultAggregation: #MAX:** encuentra el valor máximo en un campo específico. Se utiliza para identificar el valor más alto en un conjunto de datos numéricos, lo que es útil para entender los extremos superiores de los datos.
- **@DefaultAggregation: #MIN:** encuentra el valor mínimo en un campo específico. Se utiliza para identificar el valor más bajo en un conjunto de datos numéricos, permitiendo comprender los extremos inferiores de los datos.
- **@DefaultAggregation: #NONE:** indica que no se realizará ninguna agregación por defecto en el campo especificado. Se utiliza cuando no se requiere ninguna operación de agregación para ese campo, manteniendo los datos en su forma original.
- **@DefaultAggregation: #SUM:** calcula la suma total de los valores en un campo específico. Es útil para obtener el total acumulado de un conjunto de datos numéricos, proporcionando una medida agregada del volumen total de los datos.



Las agregaciones indicadas a través de las herramientas analíticas no requieren el uso obligatorio de la instrucción **group by**. Ya que estas anotaciones de agregación no corresponden a agregaciones del tipo SQL.

Ejemplo:

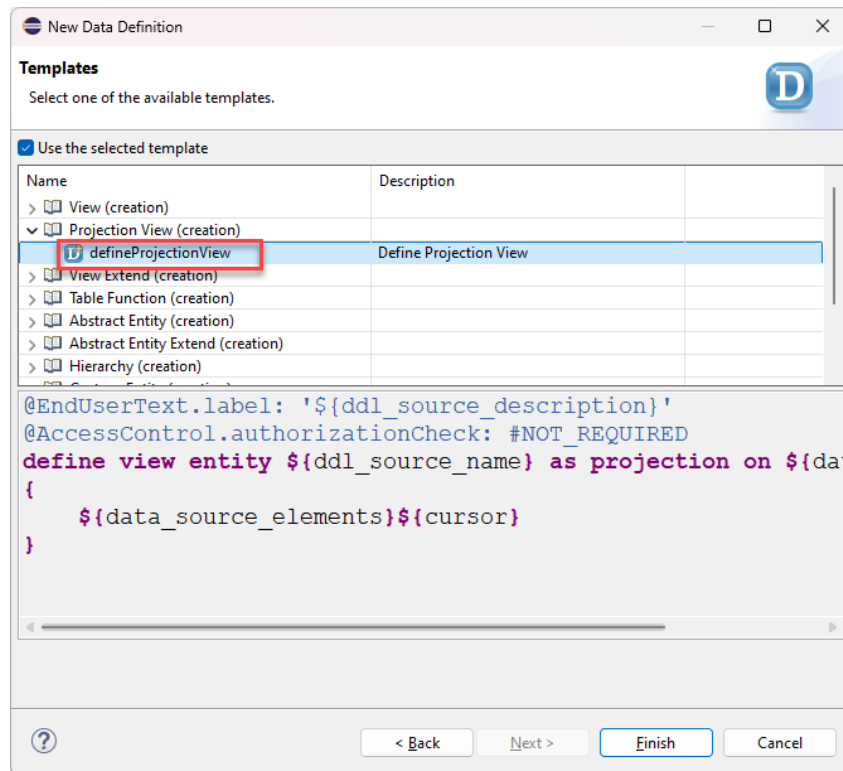
```
@AbapCatalog.viewEnhancementCategory: [#NONE]
@AccessControl.authorizationCheck: #NOT_REQUIRED
@EndUserText.label: 'CDS Virtual Data Model'
@Metadata.ignorePropagatedAnnotations: true
@ObjectModel.usageType:{
    serviceQuality: #X,
    sizeCategory: #S,
    dataClass: #MIXED
}
@Analytics.dataCategory:#CUBE
define view entity entity_name
as select from data_source_name
association [0..*] to data_source_name as _AliasName on
_AliasName .component = $projection.AliasComponentName
{
    component
    ...
    @DefaultAggregation: #SUM
    component
}
```

5.3. Transient View Entity

Transient View Entity en SAP HANA es una vista temporal y dinámica utilizada para proyectar datos analíticos en tiempo de ejecución. Estas entidades se definen utilizando Core Data Services (CDS) y se basan en otras vistas o tablas existentes. Son especialmente útiles para análisis de datos, ya que permiten la generación rápida de vistas sin necesidad de almacenamiento permanente. Las Transient View Entities se integran fácilmente con herramientas analíticas como SAP Analytics Cloud, facilitando la creación de dashboards y reportes para la toma de decisiones basada en datos en tiempo real.



Para crear un Transient View Entity se realiza el mismo procedimiento que para las proyecciones donde se crea una entidad o vista CDS a través de la carpeta de proyecto luego New en la opción **Other ABAP Repository Object** ubicar la carpeta **Core Data Services** y luego seleccionar **Data Definition** y seleccionar la plantilla **defineProjectionView** que se encuentra en la carpeta **Projection View (creation)**.



Para definir una entidad Transient View Entity en SAP HANA, se utiliza la anotación **Define Transit View Entity**. Esta definición se basa en otras vistas CDS con algunas categorías de datos, como un cubo de datos, y se proyecta utilizando el contrato **analytical_query**. Es importante eliminar las claves primarias de la vista proyectada, ya que no se permiten en este tipo de entidades.

El control de acceso para una vista entidad Transient View Entity se configura a través de la anotación **@AccessControl.authorizationCheck: #NOT_ALLOWED**

Las anotaciones analíticas:



Se utilizan para definir cómo se representarán y proyectarán los datos en las herramientas analíticas. Por ejemplo, la anotación **@analytics_details** permite definir detalles de consulta y ejes para la representación gráfica de los datos. También se pueden definir fórmulas y conversiones de datos para ajustar la visualización de la información. A continuación se describen las anotaciones analíticas:

- **@AnalyticsDetails.query.axis: #FREE:** Indica que el eje es libre, permitiendo a los usuarios finales decidir cómo utilizar los datos en el eje, proporcionando flexibilidad en la visualización y análisis.
- **@AnalyticsDetails.query.axis: #ROWS:** Asigna los datos al eje de filas, organizando y visualizando datos en un formato tabular donde las filas representan diferentes registros o elementos.
- **@AnalyticsDetails.query.axis: #COLUMNS:** Asigna los datos al eje de columnas, organizando y visualizando datos en un formato tabular donde las columnas representan diferentes atributos o medidas.

También existen algunas otras anotaciones que pueden ser utilizadas en los campos o componentes de la entidad dependiendo del caso como:

- **@ObjectModel.text.element: ['CurrencyDescription']:** Asocia un campo de texto, en este caso 'CurrencyDescription', con un campo de datos en la vista CDS, facilitando la visualización y la interpretación de los datos al agregar descripciones textuales.
- **@Aggregation.default: #FORMULA:** Define una fórmula personalizada como el método de agregación predeterminado para un campo específico, permitiendo realizar cálculos complejos o específicos en los datos. Ejemplo: conversiones del tipo de datos de una componente u operaciones dentro de la misma.

Ejemplo:

@EndUserText.label: 'CDS Transient View Entity'



```
@AccessControl.authorizationCheck: #NOT_ALLOWED
define transient view entity entity_name
provider contract analytical_query
as projection on cds_name_with_data_category
{
  @AnalyticsDetails.query.axis: #FREE
  component,
  @AnalyticsDetails.query.axis: #ROWS
  component,
  @AnalyticsDetails.query.axis: #COLUMNS
  component,
  @ObjectModel.text.element: [ 'CurrencyCode' ]
  component
}
```

5.4. Localized

La localización en SAP HANA permite ajustar y proyectar datos en diferentes idiomas y formatos regionales, asegurando que sean accesibles y comprensibles para usuarios globales. La gestión de la cardinalidad controla la relación entre conjuntos de datos, garantizando proyecciones precisas y manejables. Estas técnicas son esenciales para la proyección eficiente y correcta de datos analíticos, facilitando la toma de decisiones basada en datos en un entorno global.

Utilizando la anotación localize, es posible inyectar la clave del idioma en las proyecciones de datos, reduciendo la cardinalidad de asociaciones que podrían tener múltiples descripciones en diferentes idiomas. Esto asegura que los datos proyectados sean relevantes y específicos para el idioma del usuario.

Sintaxis:

```
_AssociationName._AssociationName2.component as  
AliasName:localized
```



5.5. Virtual Element

Los elementos virtuales en SAP HANA son campos calculados definidos temporalmente en una vista CDS (Core Data Services). Estos elementos no existen físicamente en la base de datos, sino que se generan en tiempo de ejecución, permitiendo realizar análisis y proyecciones de datos sin modificar la estructura subyacente. Los elementos virtuales facilitan la extensión de capacidades analíticas y permiten realizar cálculos complejos y conversiones de datos en tiempo real, mejorando la flexibilidad y precisión del análisis. En conjunto con la anotación **@Aggregation.default: #FORMULA**

Sintaxis:

virtual field_name : data_element_type

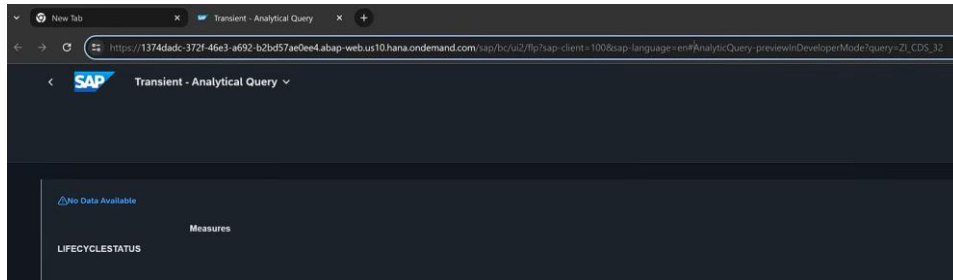
5.6. Analytic Query – Developer Mode

Las consultas analíticas en SAP HANA permiten realizar análisis detallados y dinámicos de los datos almacenados en la base de datos. Utilizando entidades de vista transitorias (Define Transit View entity), estas consultas proyectan y analizan datos sin necesidad de almacenarlos permanentemente. Son consumidas a través de herramientas analíticas avanzadas, facilitando la creación de dashboards y reportes personalizados para la toma de decisiones basada en datos. Pueden ser ejecutadas

Es posible utilizar la herramienta Analytic_Query para realizar consultas y visualizar los resultados en modo de desarrollo (Developer Mode) en el entorno de SAP Fiori. Mediante el uso de la tecla F8, ya que abre la URL de la aplicación en el navegador, accediendo a la aplicación Fiori en el dashboard a través de la cuenta en btp. Permitiendo acceder a la visualización y análisis de las consultas definidas.



Visualización de la aplicación:



Al ejecutar la entidad se puede visualizar la información que se encuentre cargada en la tabla de origen. Además habilita la herramienta developer mode que es una herramienta crucial en SAP HANA para la visualización y prueba de consultas analíticas (Analytic Queries). Este modo permite a los desarrolladores obtener una vista previa de cómo se comportan sus consultas en un entorno real sin necesidad de realizar implementaciones completas. Entre las herramientas tenemos Filtros, Layouts(Char Settings), Formateos, información del Contexto y la posibilidad de exportar a través de un archivo pdf o excel.

