Хранилище данных предназначено для анализа информации о сотрудниках компании. Оно содержит данные о:

* **Сотрудниках (FactEmployee):** ID сотрудника, ID отдела, ID должности, зарплата.
* **Отделах (DimDepartment):** ID отдела, название отдела, местоположение (город).
* **Должностях (DimPosition):** ID должности, название должности, грейд зарплаты (уровень).

Бизнес-задачи, которые можно решить с текущей структурой хранилища:

1. **Расчёт затрат на зарплаты:**
   * Общая сумма зарплат по компании.
   * Сумма зарплат по каждому отделу.
   * Сумма зарплат по каждой должности.
   * Средняя зарплата по отделам и должностям.
2. **Анализ численности персонала:**
   * Общее количество сотрудников.
   * Количество сотрудников в каждом отделе.
   * Количество сотрудников на каждой должности.
3. **Анализ по местоположению (если актуально для компании)**
   * Сумма затрат по каждому местоположению (городу)
   * Количество сотрудников в каждом городе.
   * Средняя зарплата по городам.

**Таблица фактов (Fact Table):** FactEmployee

* Это центральная таблица, содержащая числовые данные (меры), которые мы хотим анализировать.
* Она связана с таблицами измерений через внешние ключи (Foreign Keys).
* Таблица фактов содержит информацию о зарплате каждого сотрудника, связанную с его отделом и должностью.

**Измерения (Dimensions):**

* Измерения — это, по сути, справочники, которые содержат атрибуты, описывающие данные в таблице фактов. Они позволяют отвечать на вопросы "кто?", "что?", "где?", "когда?".
* DimDepartment: Предоставляет контекст для анализа данных о сотрудниках. Позволяет группировать и фильтровать данные по отделам. Содержит атрибуты:
  + DepartmentID (первичный ключ)
  + DepartmentName
  + Location
* DimPosition: Предоставляет контекст для анализа по должностям. Содержит атрибуты:
  + PositionID (первичный ключ)
  + PositionName
  + SalaryGrade

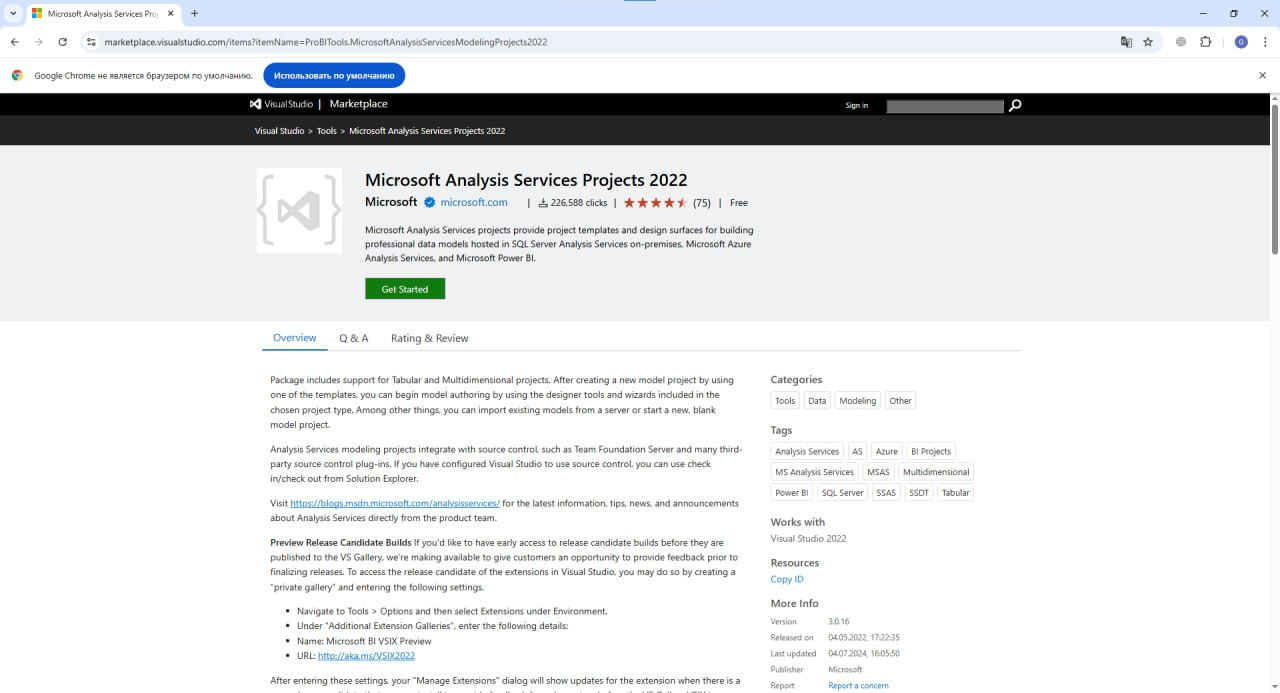
|  |
| --- |
| CREATE DATABASE HRDataWarehouse;  USE HRDataWarehouse;  -- Таблица фактов  CREATE TABLE FactEmployee (  EmployeeID INT PRIMARY KEY,  DepartmentID INT,  PositionID INT,  Salary DECIMAL(10, 2)  );  -- Таблица измерений  CREATE TABLE DimDepartment (  DepartmentID INT PRIMARY KEY,  DepartmentName NVARCHAR(100),  Location NVARCHAR(100)  );  CREATE TABLE DimPosition (  PositionID INT PRIMARY KEY,  PositionName NVARCHAR(100),  SalaryGrade NVARCHAR(50)  );  ALTER TABLE FactEmployee  ADD CONSTRAINT FK\_FactEmployee\_DimDepartment  FOREIGN KEY (DepartmentID) REFERENCES DimDepartment(DepartmentID);  ALTER TABLE FactEmployee  ADD CONSTRAINT FK\_FactEmployee\_DimPosition  FOREIGN KEY (PositionID) REFERENCES DimPosition(PositionID);  INSERT INTO DimDepartment (DepartmentID, DepartmentName, Location)  VALUES (1, 'Sales', 'Moscow'),  (2, 'HR', 'St. Petersburg');  INSERT INTO DimPosition (PositionID, PositionName, SalaryGrade)  VALUES (1, 'Manager', 'High'),  (2, 'Analyst', 'Medium');  INSERT INTO FactEmployee (EmployeeID, DepartmentID, PositionID, Salary)  VALUES (1, 1, 1, 100000),  (2, 2, 2, 80000);    SELECT \* FROM DimDepartment;  SELECT \* FROM DimPosition;  SELECT \* FROM FactEmployee; |

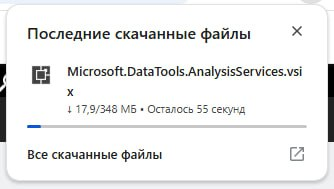
**Меры (Measures):**

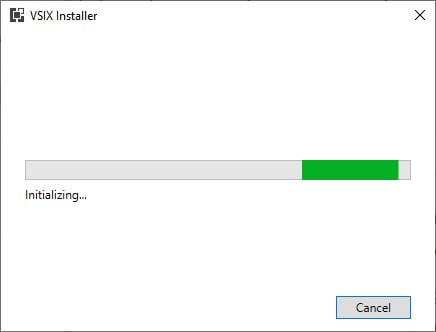
* Salary (из таблицы FactEmployee): Это числовое значение (сумма денег), которое мы будем анализировать. Меры — это то, *что* мы измеряем. Типичные операции с мерами — это суммирование (SUM), вычисление среднего (AVG), нахождение минимума (MIN) и максимума (MAX).
* *Неявная мера*: Во многих инструментах OLAP, автоматически создается неявная мера Count (или [FactEmployee Count]), которая подсчитывает количество записей в таблице фактов (т.е. количество сотрудников).

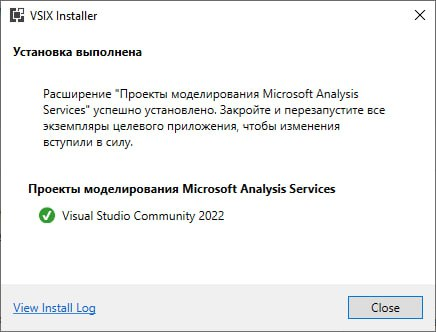
SQL Server Analysis Services (SSAS) — это служба аналитики и платформа для построения хранилищ данных и бизнес-аналитики от Microsoft.

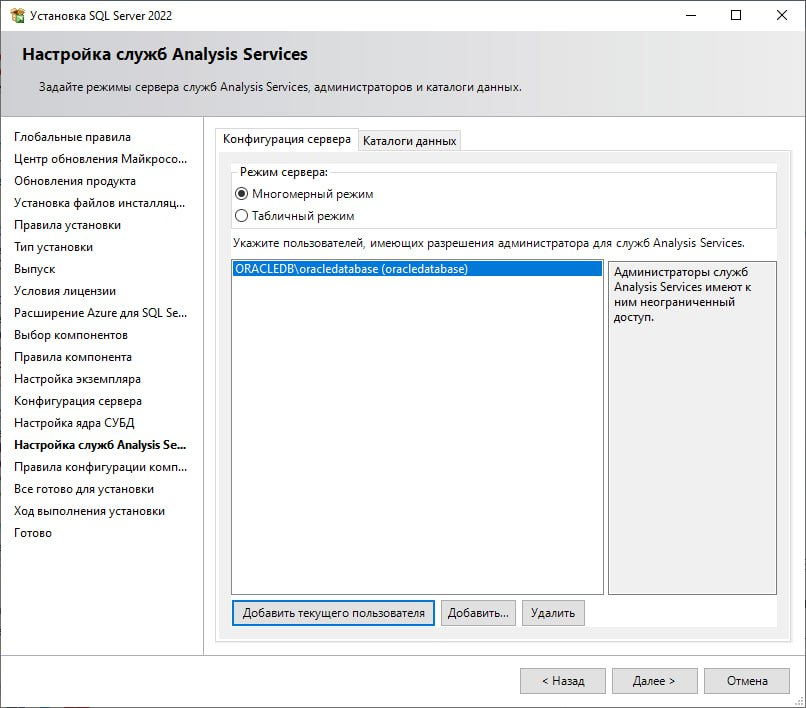
* SSAS позволяет создавать многомерные OLAP-кубы (Online Analytical Processing). Куб — это структура данных, оптимизированная для быстрого анализа больших объёмов информации. Данные в кубе предварительно агрегируются (суммируются, усредняются и т.д.), что значительно ускоряет выполнение запросов.
* Для запросов к кубам используется язык MDX (Multidimensional Expressions). Он отличается от SQL и предназначен для работы с многомерными данными.

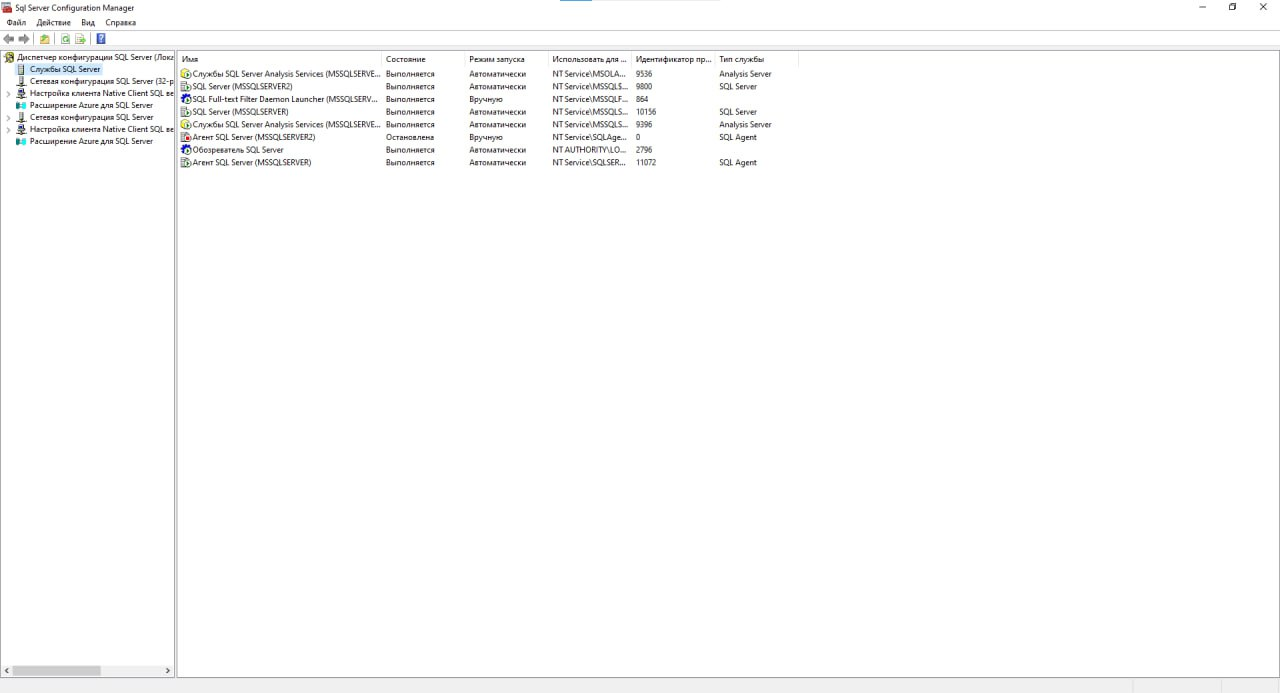


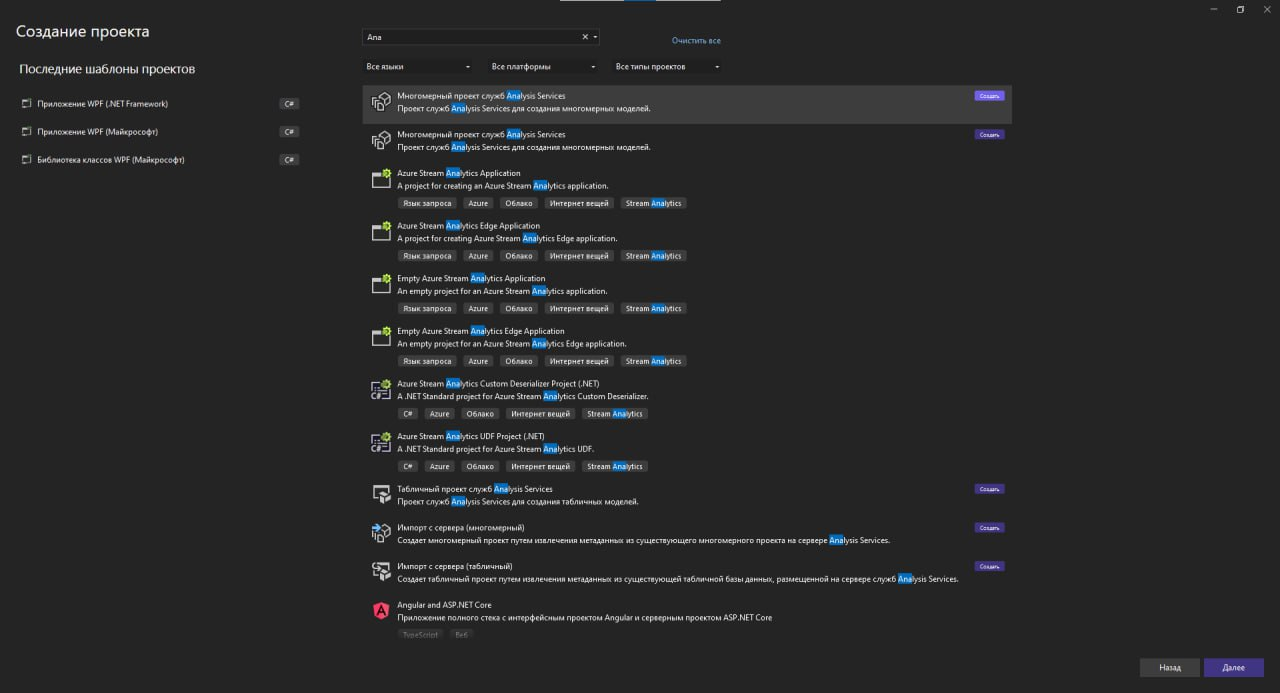


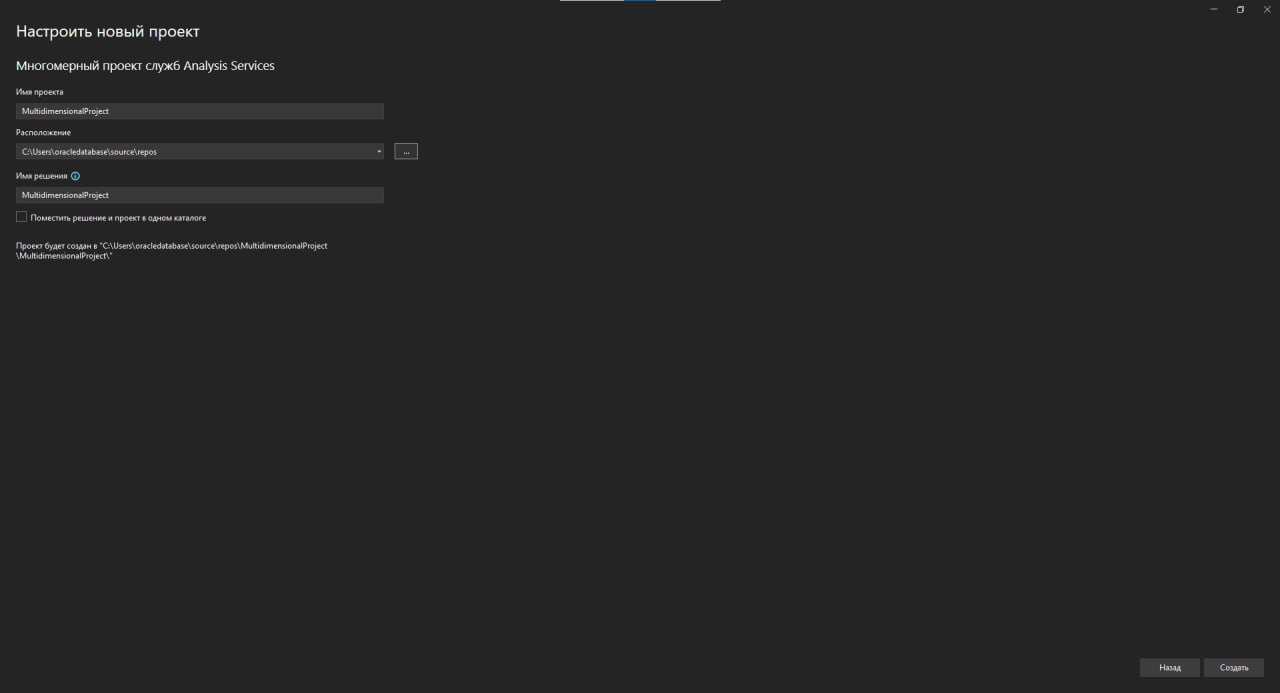




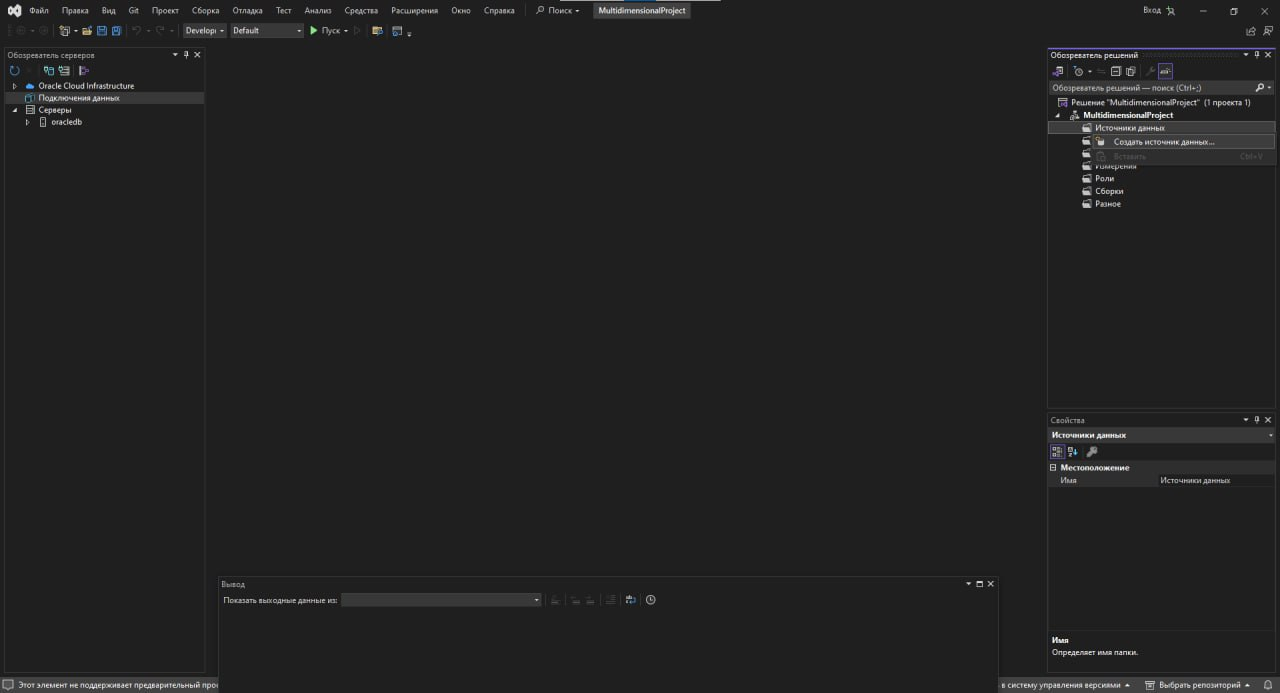




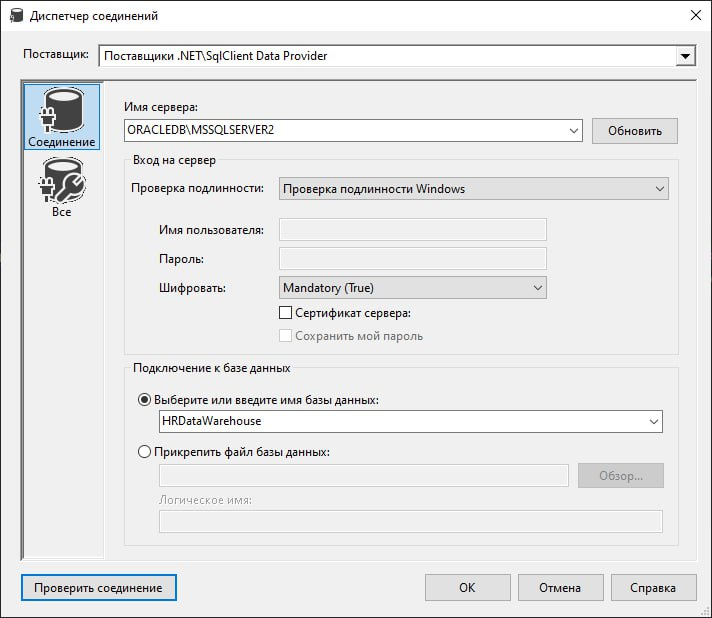


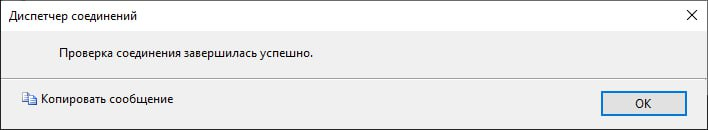


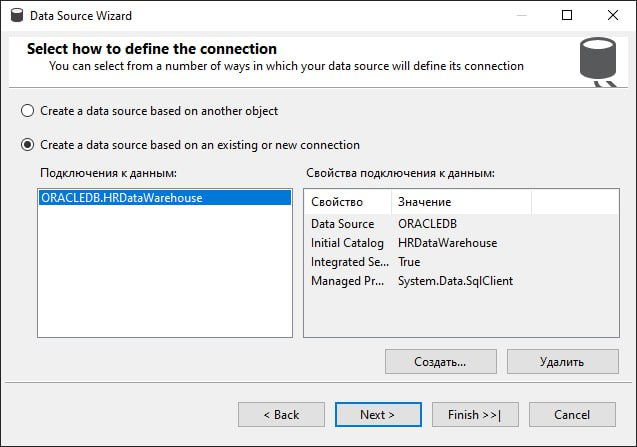
**Источник данных (Data Source)** – это *соединение* с базой данных (или другим источником данных, например, текстовым файлом), из которой SSAS будет получать данные для куба.

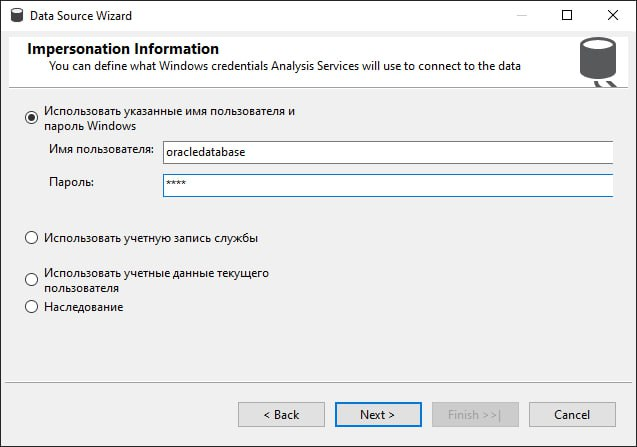


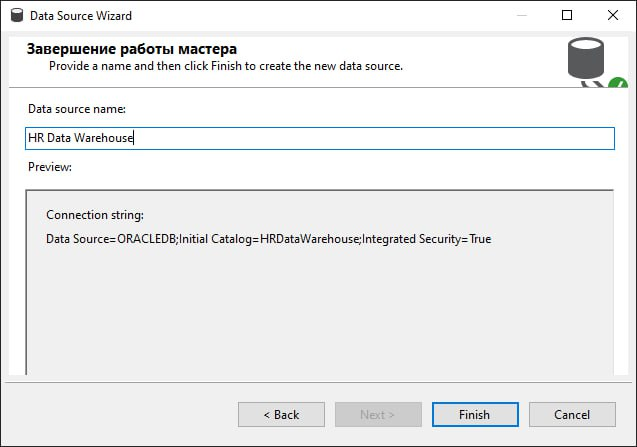




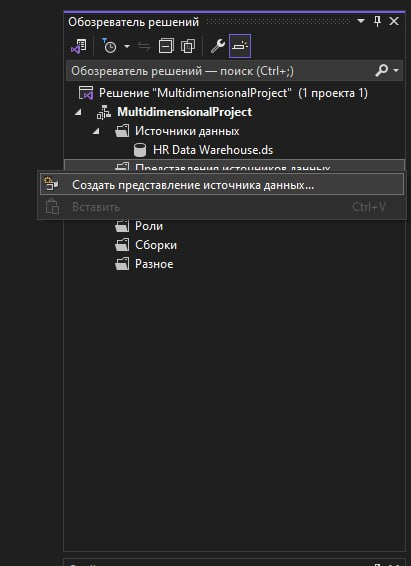


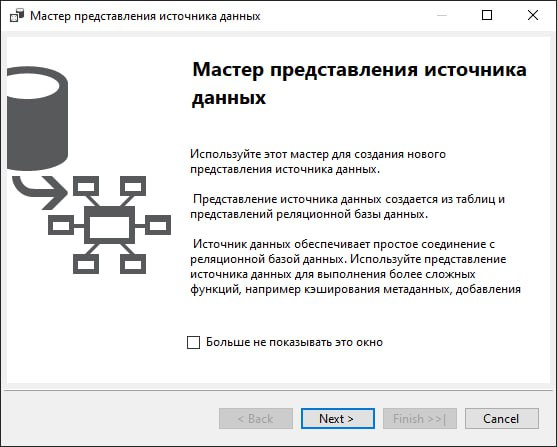


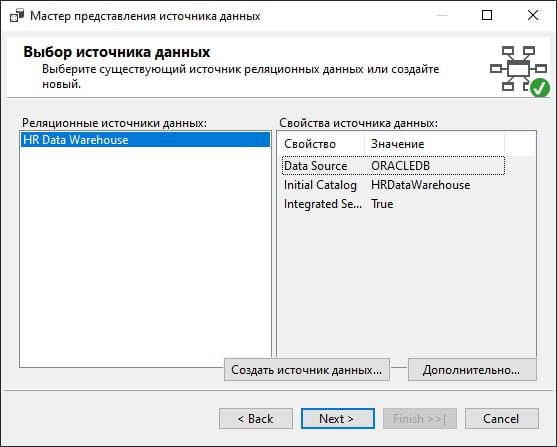


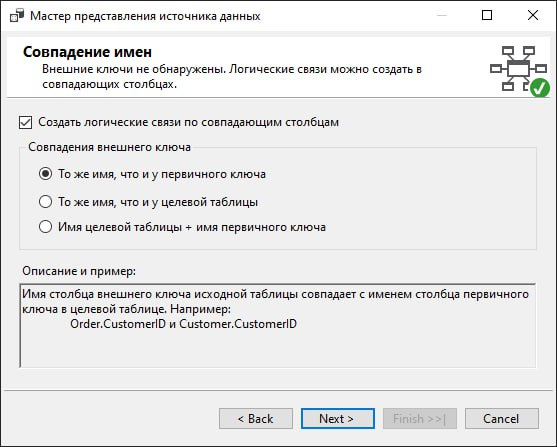


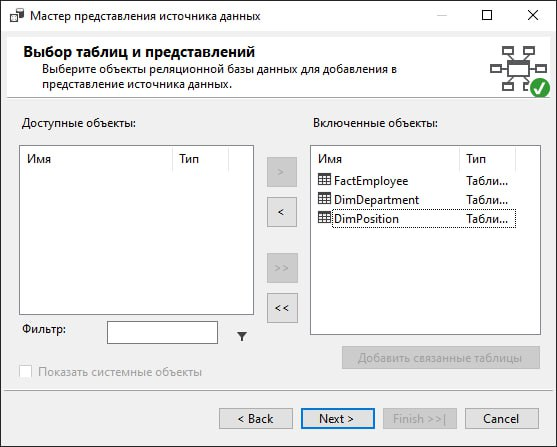
**Представление источника данных (Data Source View, DSV) –** это *абстрактное представление* данных из одного или нескольких источников данных. Это *логический слой* между источником данных и кубом.

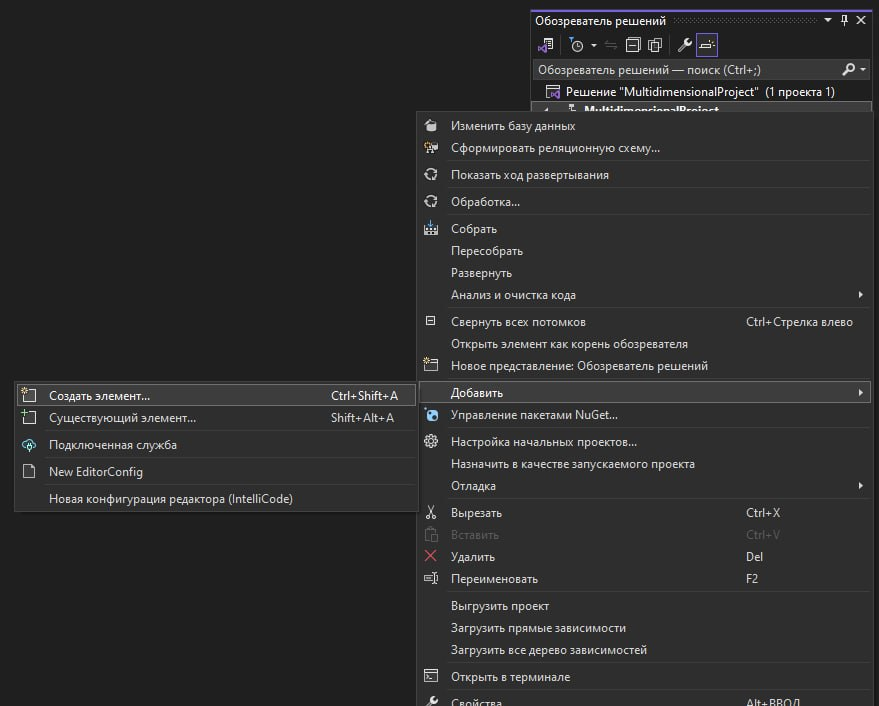


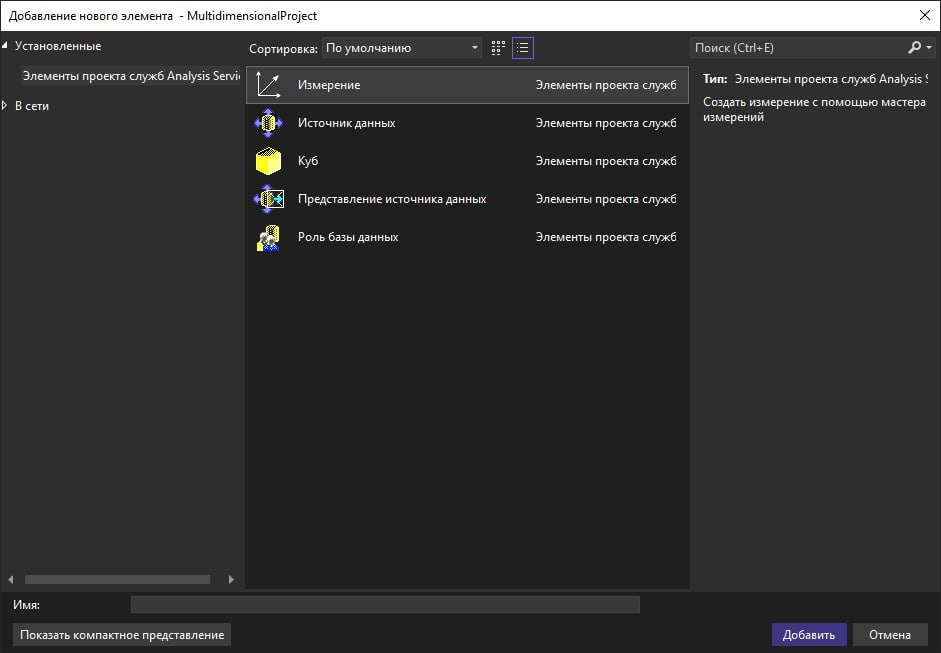


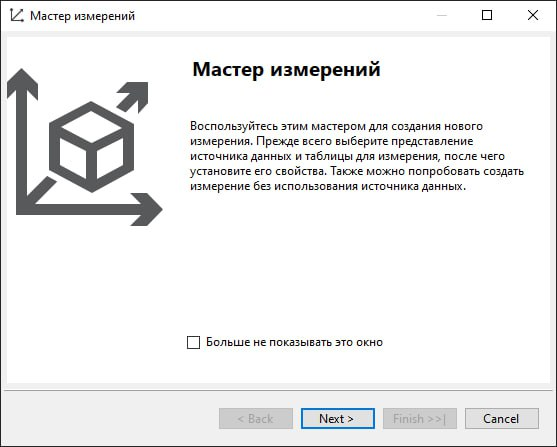


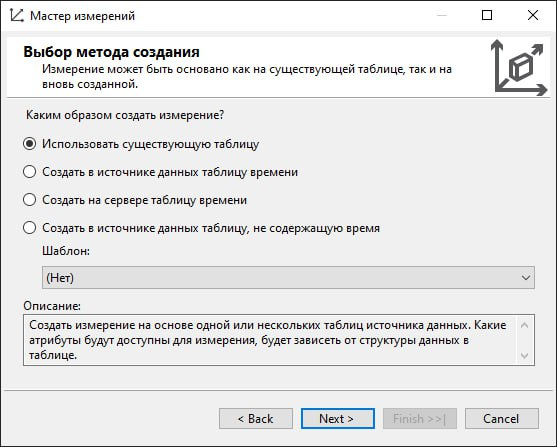


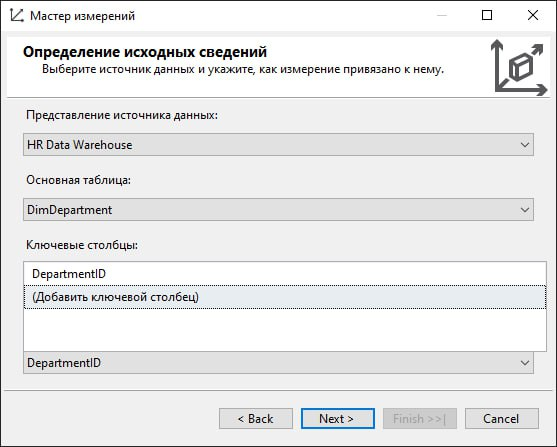


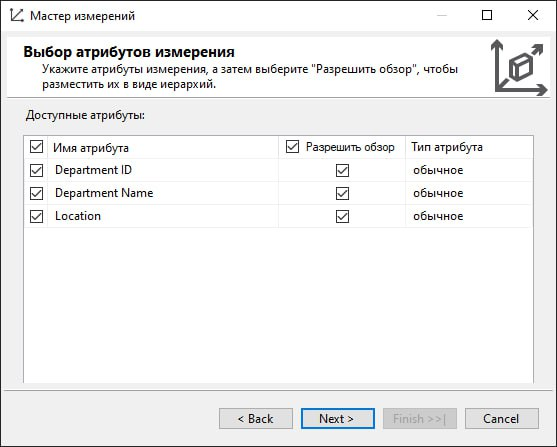


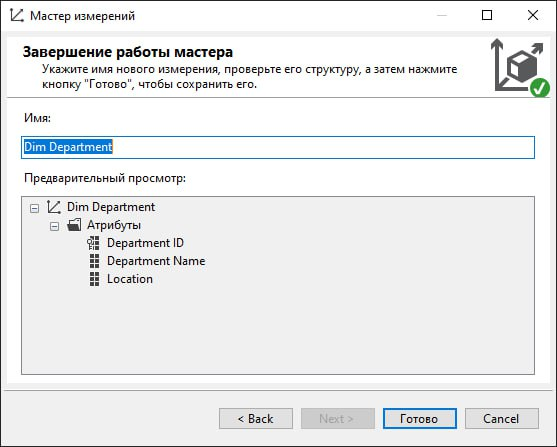


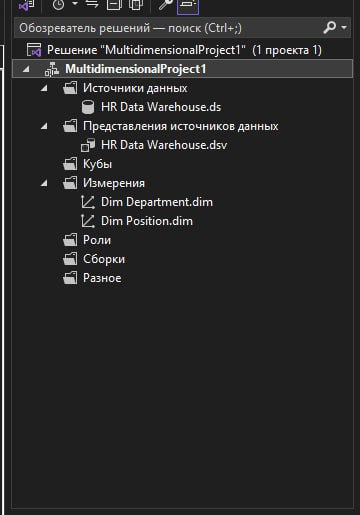


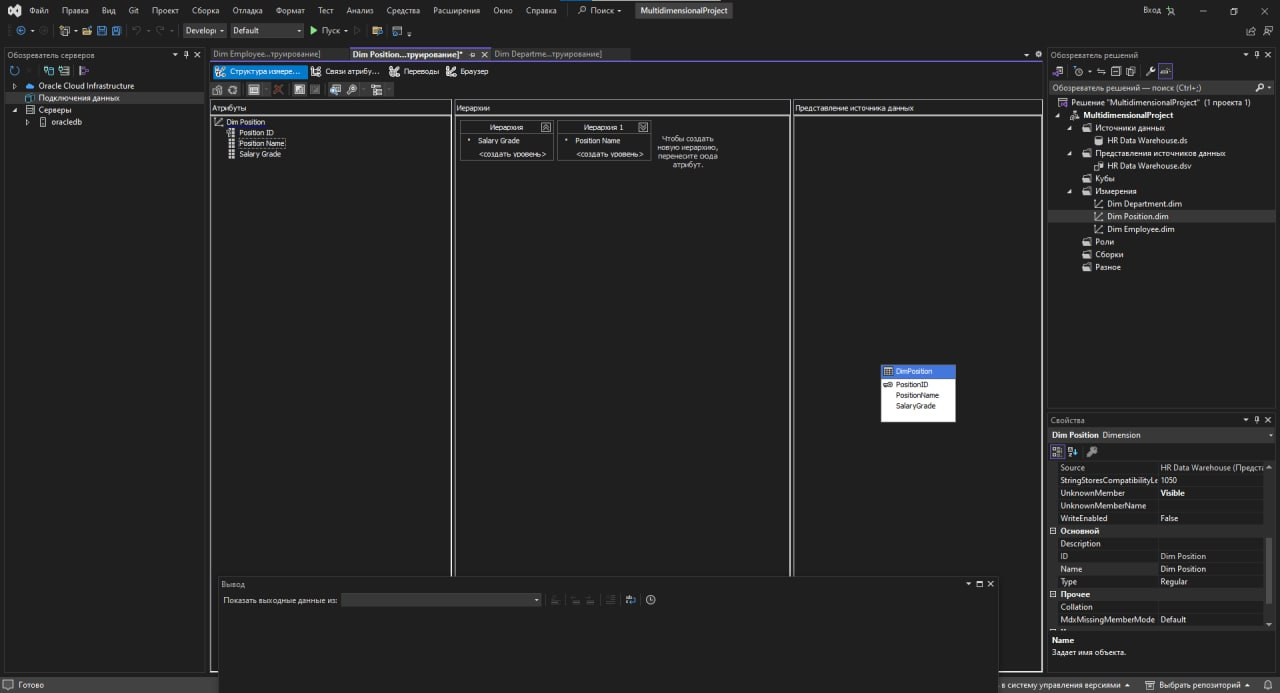


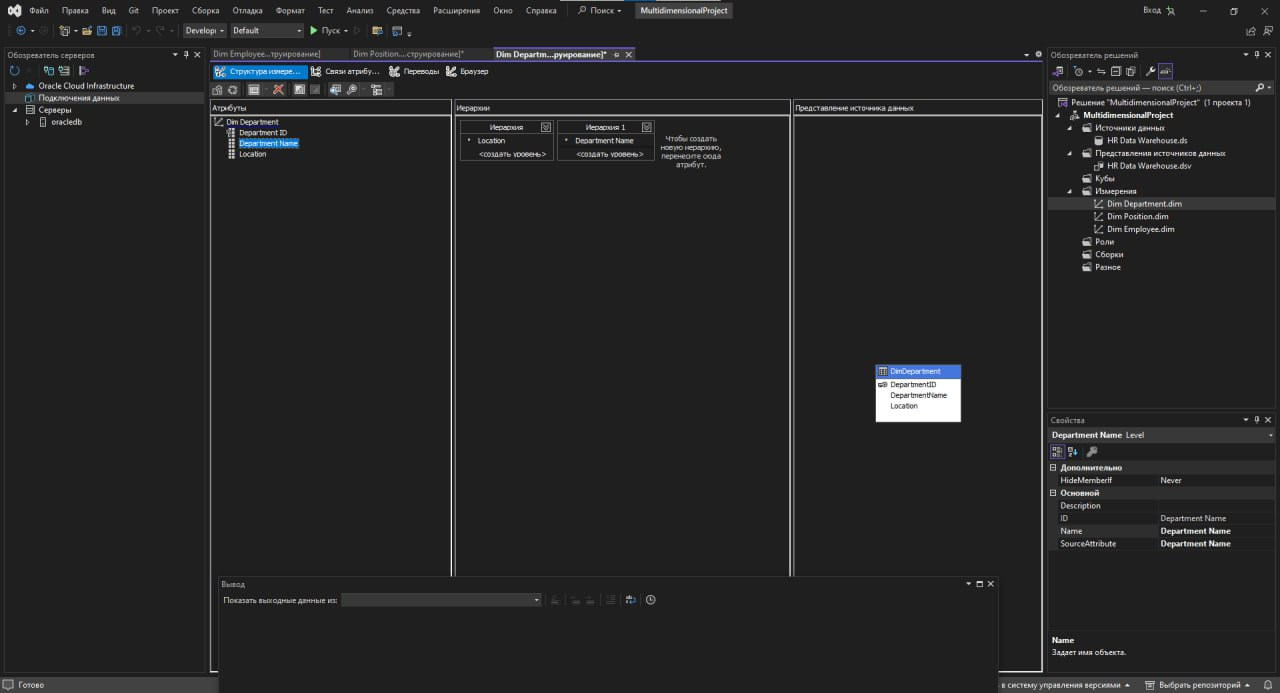


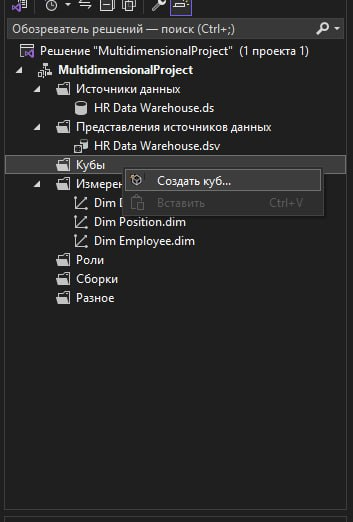


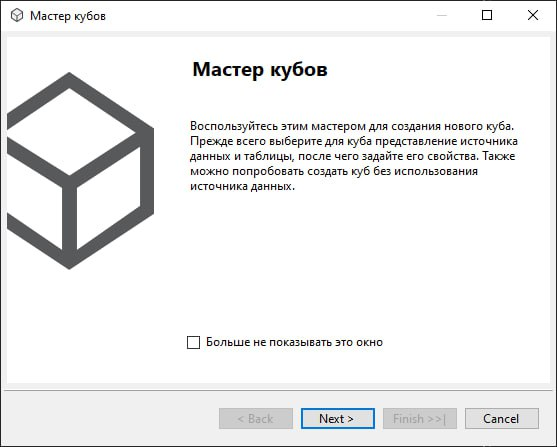


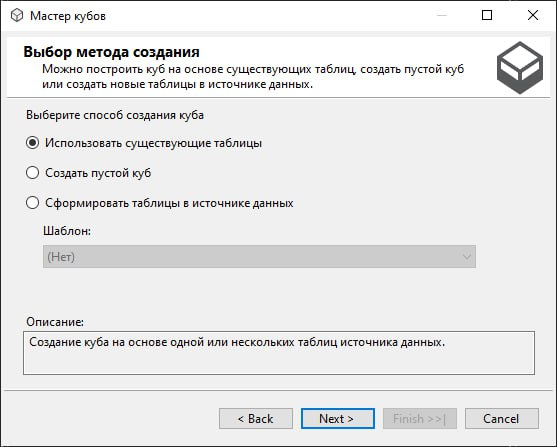


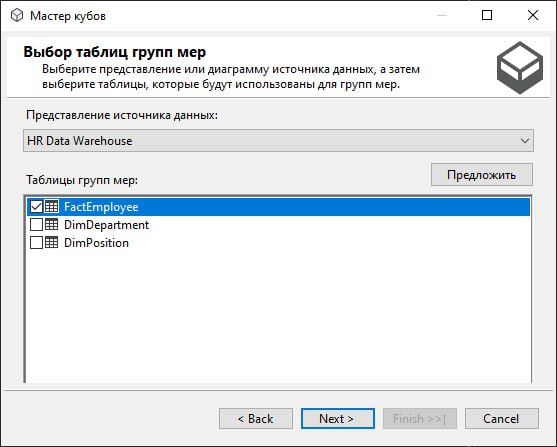


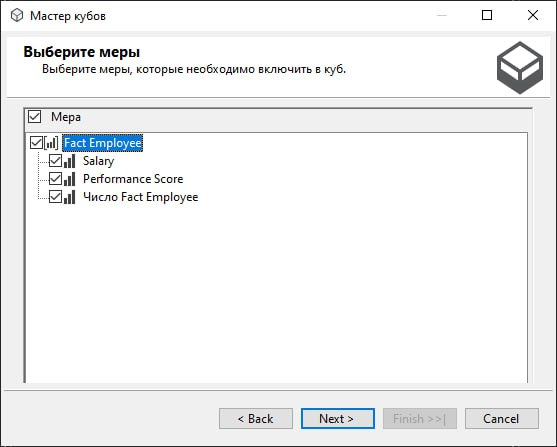


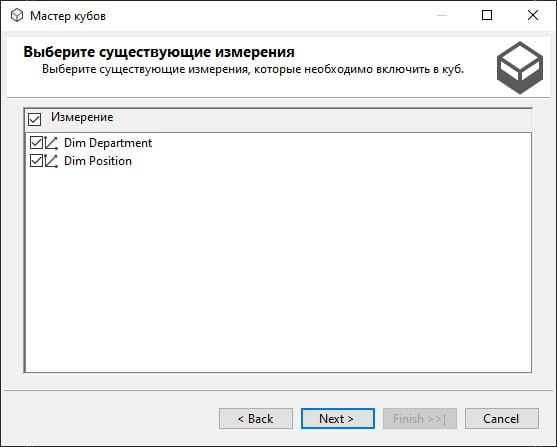


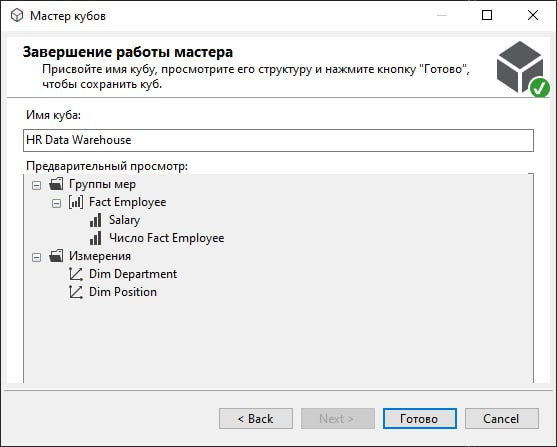


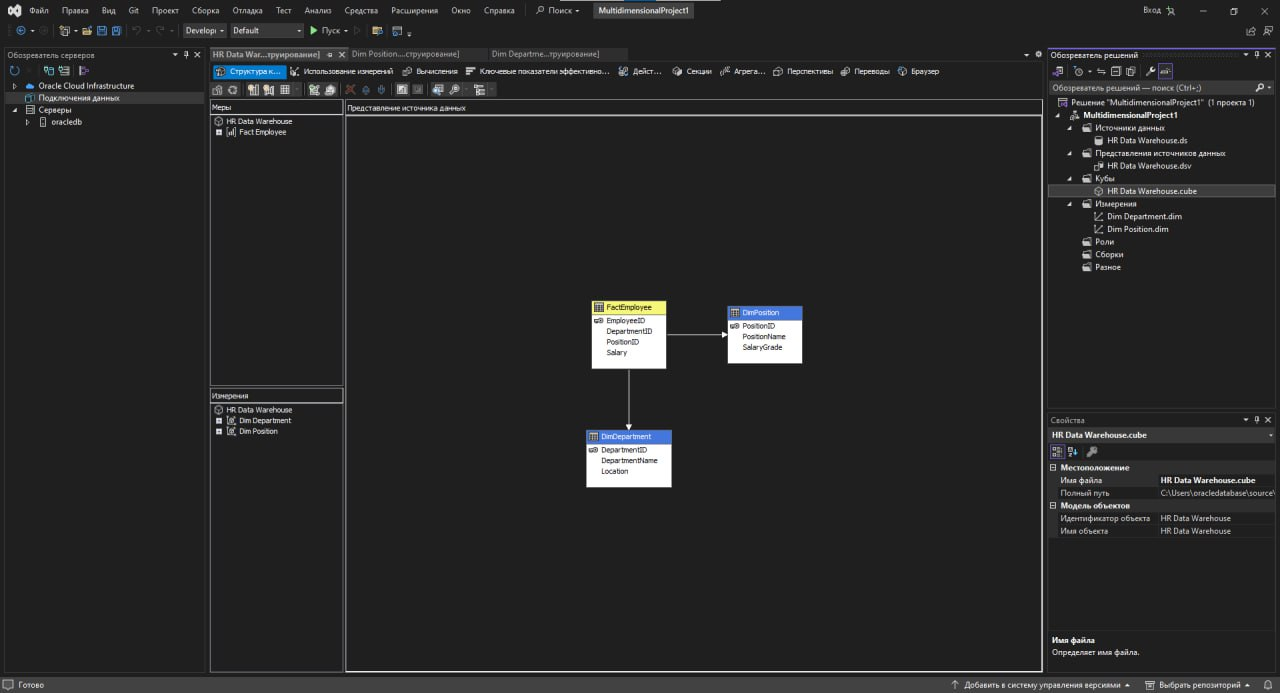






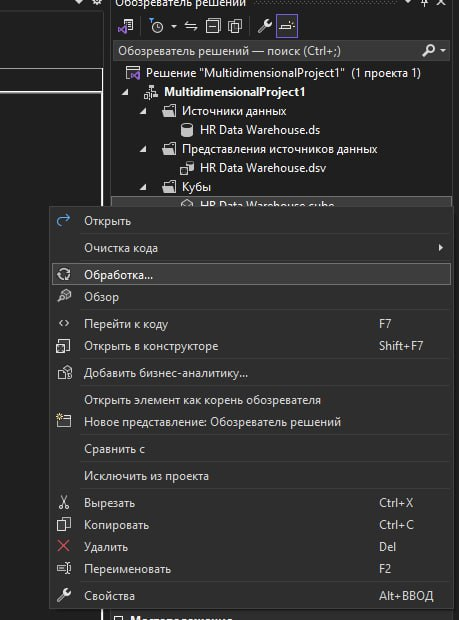


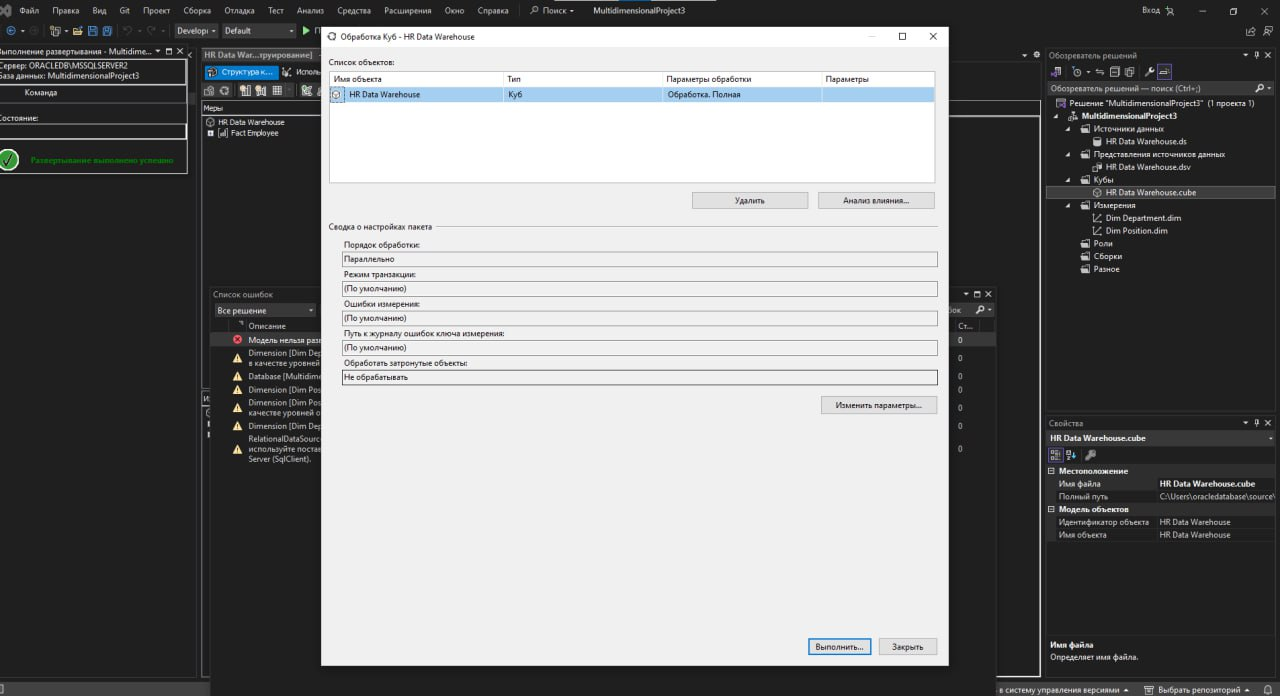


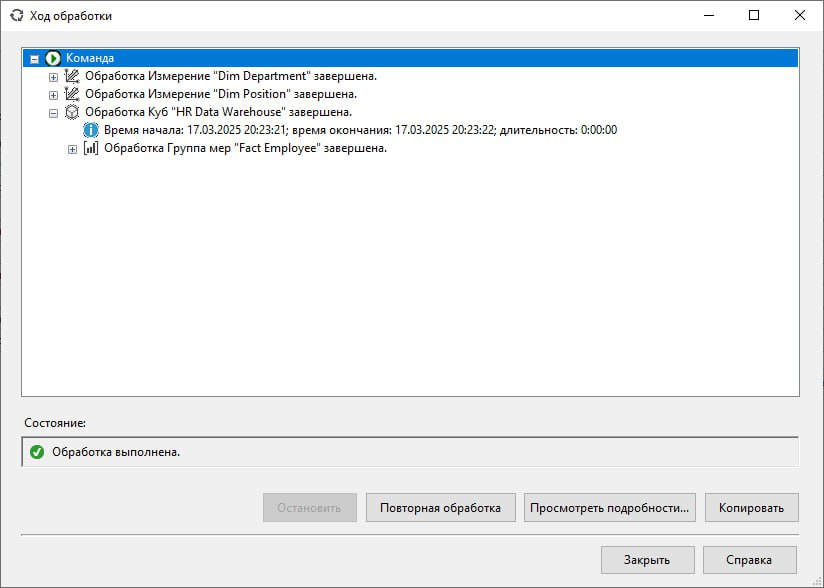


После создания структуры куба (определения измерений, мер, иерархий) в SQL Server Analysis Services (SSAS), его необходимо обработать (Process).

Обработка куба (Processing) – это процесс загрузки данных из источника данных (через представление источника данных) в многомерную структуру куба и выполнения необходимых вычислений.



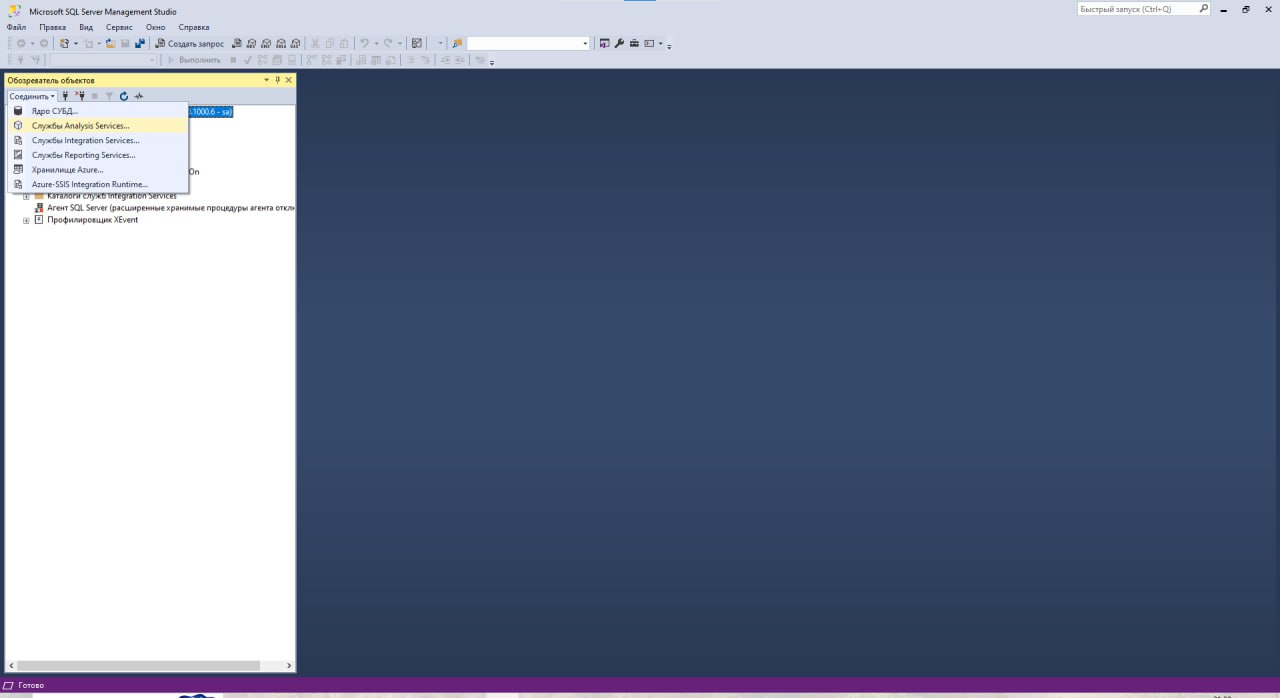


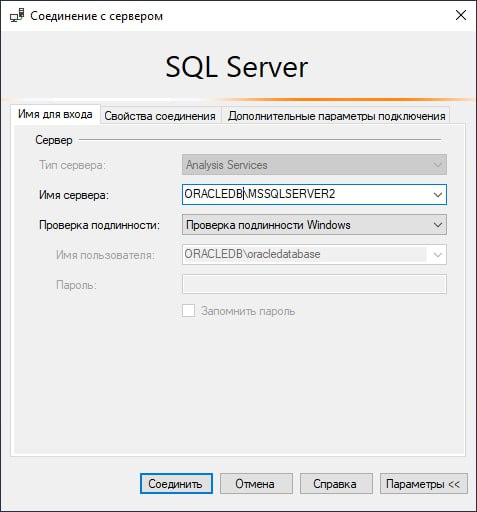


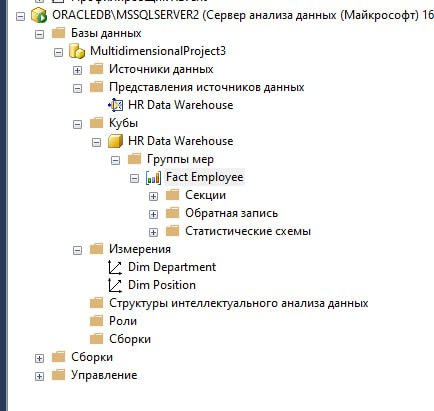
Мы работаем с *проектом* Analysis Services. Этот проект содержит определения куба (источники данных, DSV, измерения, меры, иерархии и т.д.).

Чтобы куб стал доступен для использования, его нужно *развернуть* (deploy) на экземпляре (instance) Analysis Services.

При развертывании мы указываем *сервер* SSAS, на котором нужно создать куб, и *базу данных* Analysis Services, в которой он будет размещен.







* 1. Числовые;

Выполнение математических операций над числовыми данными (мерами) в кубе. Позволяют вычислять суммы, средние значения, минимумы, максимумы, стандартные отклонения и многое другое.

WITH MEMBER [Measures].[Average Salary] AS

Avg([Dim Department].[Department Name].MEMBERS, [Measures].[Salary])

SELECT

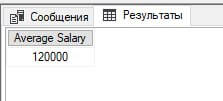
{ [Measures].[Average Salary] } ON COLUMNS

FROM [HR Data Warehouse];

* **WITH MEMBER [Measures].[Average Salary] AS ...**: Создание *вычисляемой меры* (calculated member) с именем [Measures].[Average Salary]. Вычисляемые меры – это временные показатели, которые существуют только во время выполнения запроса.
* **Avg([Dim Department].[Department Name].MEMBERS, [Measures].[Salary])**:
  + Avg(...): Это функция MDX, которая вычисляет *среднее значение*.
  + [Dim Department].[Department Name].MEMBERS: Это *набор всех отделов*. .MEMBERS возвращает все элементы уровня "Department Name" измерения "Dim Department".
  + [Measures].[Salary]: Это мера "Salary" (зарплата).

Вычислить среднее значение меры Salary по *всем отделам*. То есть, функция Avg просуммирует зарплаты всех сотрудников во всех отделах и разделит на общее количество сотрудников.

* **SELECT { [Measures].[Average Salary] } ON COLUMNS**:
  + SELECT: Указывает, какие данные нужно выбрать из куба.
  + { [Measures].[Average Salary] }: Выбираем нашу вычисляемую меру [Measures].[Average Salary]. Фигурные скобки {} создают набор (в данном случае, из одного элемента).
  + ON COLUMNS: Размещаем этот набор (нашу меру) на оси *столбцов* результата.
* **FROM [HR Data Warehouse]**:
  + Указывает, из какого *куба* нужно выбрать данные (HR Data Warehouse).



* 1. Обработки строковых значений;

Работа со строковыми значениями атрибутов измерений. Позволяют извлекать подстроки, конкатенировать строки, преобразовывать регистр и т.д.

WITH MEMBER [Measures].[DepartmentShortName] AS

Left([Dim Department].[Department Name].CurrentMember.Name, 3)

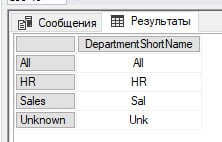
SELECT

{ [Measures].[DepartmentShortName] } ON COLUMNS,

{ [Dim Department].[Department Name].MEMBERS } ON ROWS

FROM [HR Data Warehouse];

* **WITH MEMBER:** Это ключевая конструкция MDX, которая позволяет определять *вычисляемые элементы* (calculated members) *внутри* запроса.
* **[Measures].[DepartmentShortName]:** Это *имя* нового вычисляемого элемента, который мы создаем. Мы добавляем его к измерению [Measures], то есть это будет новая мера.
* **AS:** Ключевое слово, которое указывает, что дальше идет *определение* вычисляемого элемента.
* **Left([Dim Department].[Department Name].CurrentMember.Name, 3):** Это *выражение*, которое вычисляет значение новой меры [Measures].[DepartmentShortName].
  + **Left(...):** Это *функция* MDX, которая работает со строками. Она возвращает указанное количество символов *с начала* строки. **Функция Left принимает два аргумента:**
    1. **Строка, из которой нужно извлечь символы.**
    2. **Количество символов, которое нужно извлечь.**
  + **[Dim Department].[Department Name].CurrentMember:** Это *ключевая* часть выражения. Она ссылается на *текущий* элемент уровня "Department Name" измерения "Dim Department".
  + **.Name:** Это *свойство* элемента измерения. Оно возвращает *имя* элемента (строковое значение). То есть, если CurrentMember указывает на отдел "Sales", то CurrentMember.Name вернет строку "Sales".
  + **, 3:** Второй аргумент функции Left – число 3. Это значит, что мы хотим извлечь *первые три* символа из имени отдела.



* 1. Для наборов;

Работа с *наборами* (множествами) элементов измерений. Позволяют фильтровать, сортировать, объединять, пересекать наборы, получать подмножества (например, TopCount) и т.д. *Очень важная* категория функций для MDX.

SELECT

{ [Measures].[Salary] } ON COLUMNS,

Filter(

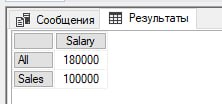
[Dim Department].[Department Name].MEMBERS,

[Measures].[Salary] > 90000

) ON ROWS

FROM [HR Data Warehouse]

* **SELECT**: Выбор данных.
* **{ [Measures].[Salary] } ON COLUMNS**: Мера Salary на оси столбцов.
* **Filter(...) ON ROWS**: Здесь используется функция Filter для работы с набором.
* **Filter(set, condition):** Эта функция принимает два аргумента:
  + set: Набор элементов, который нужно отфильтровать.
  + condition: Логическое условие, которое применяется к каждому элементу набора. Функция возвращает *новый набор*, содержащий только те элементы, для которых условие истинно.
* [Dim Department].[Department Name].MEMBERS: Это набор всех названий отделов (как в предыдущем запросе).
* [Measures].[Salary] > 90000: Это условие фильтрации. Для *каждого отдела* проверяется, больше ли общая сумма зарплат по этому отделу, чем 90000.
* ON ROWS\*\*: Результат работы функции Filter (набор отделов, удовлетворяющих условию) размещается на оси строк.



* 1. Для элемента измерения;

Работа с отдельными элементами измерений. Позволяют получать имя элемента, его родителя, потомков, уровень в иерархии, свойства элемента и т.д.

WITH MEMBER [Measures].[DepartmentShortName] AS

Left([Dim Department].[Department Name].CurrentMember.Name, 3)

SELECT

{ [Measures].[DepartmentShortName] } ON COLUMNS,

{ [Dim Department].[Department Name].MEMBERS } ON ROWS

FROM [HR Data Warehouse];

* 1. Для иерархий;

WITH MEMBER [Measures].[HierarchyLevelCount] AS

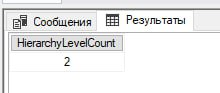
[Dim Department].[Department Name].Levels.Count

SELECT

{ [Measures].[HierarchyLevelCount] } ON COLUMNS

FROM [HR Data Warehouse];

* **WITH MEMBER [Measures].[HierarchyLevelCount] AS ...**: Создает *временную вычисляемую меру* с именем [Measures].[HierarchyLevelCount]. Эта мера будет существовать только во время выполнения запроса.
* **[Dim Department].[Department Name].Levels.Count**:
  + [Dim Department].[Department Name]: Обращение к атрибуту "Department Name" измерения "Dim Department".
  + .Levels: Обращение к *коллекции уровней* иерархии, связанной с этим атрибутом.
  + .Count: *Свойство* коллекции уровней, которое возвращает *количество* уровней в этой иерархии.
* **SELECT { [Measures].[HierarchyLevelCount] } ON COLUMNS**:
  + Выбирает созданную меру [Measures].[HierarchyLevelCount] и размещает её на оси столбцов результата (т.е. результатом будет таблица с одним столбцом).
* **FROM [HR Data Warehouse]**:
  + Указывает, что данные нужно брать из куба HR Data Warehouse.



1. **"All" Level (Уровень "Все"):** Этот уровень создается *всегда* для любого атрибута измерения. Он содержит один-единственный элемент, который называется "All" (или "(All)"). Этот элемент представляет собой *агрегацию* всех элементов нижележащего уровня.
2. **"Department Name" Level (Уровень с названиями отделов):** Этот уровень содержит все *фактические* значения атрибута Department Name – названия отделов.