**WCF Data Services:** протокол Open Data Protocol, возможности предоставляемые OData-интерфейсом, порядок разработки Data Services, применение Data Services.

**Open Data Protocol –** открытый web-протокол; позволяет выполнять операции с ресурсами и получать ответы в форматах XML, JSON.

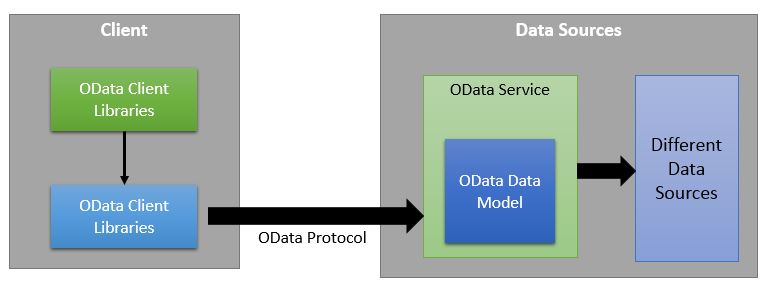
**OData:** поддерживается OASIS.

**OData:** поддерживается практически всеми системами программирования, в т.ч. JavaScript

**Архитектура:**

На диаграмме ниже поясняется высокоуровневая архитектура OData. Существует четыре основные части OData.

* Модель OData
* Служба OData
* Протокол OData
* Клиентские библиотеки OData



**Модель OData:**

Модель OData используется для представления данных из разных источников данных в едином формате с использованием модели Entity Data (EDM).

**Служба OData:**

Это уровень обслуживания поверх модели OData, который предоставляет конечную точку, которая позволяет клиенту получать доступ к данным с использованием протокола OData и клиентской библиотеки OData.

Он преобразует различные форматы источников данных, такие как таблицы отношений SQL, список SharePoint, таблицы Windows Azure и т. д., в общий формат для использования клиентом.

**Протокол OData:**

Протокол OData позволяет клиенту отправлять запросы и получать ответы от службы OData.

Протокол OData — это набор взаимодействий RESTful.

Протокол OData **аналогичен HTTP** и используется для выполнения операций CRUD с использованием OData. **Он передает данные в форме XML или JSON.**

**Клиентские библиотеки OData:**

Клиентские библиотеки OData используются клиентским приложением для доступа к данным с использованием протокола OData.

Предоставление готовых библиотек для выполнения запросов OData и получения результатов упрощает жизнь разработчикам, создающим эти приложения.

Для доступа к OData доступно несколько клиентских библиотек OData. Некоторые из них:

* Microsoft .NET Framework 3.51: **платформа WCF Data Services** доступна для отдельной загрузки для .NET 3.x.
* Java: **odata4j** (включая Java на телефоне Android) поддерживает протокол OData.
* JavaScript: библиотека **OpenUI5,** поддерживаемая **SAP.**
* И многое другое.

**Возможности интерфейса:**

1. *Фильтрация данных:*
   * http://localhost:5000/odata/Students?$**filter**=Grade eq 5 - получение студентов с оценкой 5.
2. *Сортировка данных:*
   * http://localhost:5000/odata/Students?$**orderby**=Name desc - получение студентов, отсортированных по имени в обратном порядке.
3. *Проекция данных (выбор конкретных полей):*
   * http://localhost:5000/odata/Students?$**select**=Name,Grade - получение только имен и оценок студентов.
4. *Расширенные запросы:*
   * http://localhost:5000/odata/Students?$**top**=2 - получение первых двух студентов.
   * http://localhost:5000/odata/Students?$**skip**=2 - пропуск первых двух студентов и получение остальных.
   * http://localhost:5000/odata/Students?$**count**=true - получение общего количества студентов.
5. *Комбинированные запросы:*
   * http://localhost:5000/odata/Students?$filter=Grade gt 3 and Grade lt 6&$orderby=Grade desc - получение студентов с оценкой между 3 и 6, отсортированных по убыванию оценок.
6. *Поиск:*
   * http://localhost:5000/odata/Students?$**search**=Иванов - поиск студента с именем "Иванов".
7. *Группировка данных:*
   * http://localhost:5000/odata/Students?$apply=groupby((Grade), aggregate(Name with max as MaxName)) - группировка студентов по оценке с агрегацией максимального имени в каждой группе.
8. *Inline Count (получение количества записей внутри ответа):*
   * http://localhost:5000/odata/Students?$top=2&$count=true - получение первых двух студентов и общего количества студентов.

**Порядок разработки Data Services:**

1. Создание БД
2. Создание приложения ASP.NET
3. Создание EDM модели
4. Создание WCF Data Service

**Применение Data Services:**

* простое чтение данных
* параметризированные запросы
* добавление данных
* модификация данных
* удаление данных
* добавление/изменение/удаление связей между объектами