# Унифицированный доступ к БД

## OLE DB

Разработан для доступа к более широкому спектру источников данных, как реляционных, так и нереляционных.

OLE DB (англ. Object Linking and Embedding, Database) — набор COM-интерфейсов, которые позволяют приложениям унифицировано работать с данными разных источников и хранилищ информации. Разработана Microsoft в качестве дальнейшего развития технологии доступа к данным и должен был прийти на замену и в качестве преемника ODBC, расширяя набор функций для поддержки более широкого круга нереляционных источников данных, таких как объектно-ориентированные базы данных или электронные таблицы, и для которых не обязательно использовать SQL.

OLE DB отделяет хранилище данных от приложения, которое должно иметь доступ к нему через набор абстракций, состоящий из источника данных (DataSource), сессии (Session), команды (Command) и набора строк (Rowset). Это было сделано для предоставления унифицированного доступа к различным видам и источникам данных и изоляцию специфики взаимодействия с конкретным хранилищем. OLE DB концептуально разделена на потребителей (клиентов) и поставщиков (провайдеров). Потребителем является приложение, которому необходим доступ к данным, а поставщик реализует интерфейс доступа к данным и, следовательно, обеспечивает информацией потребителя.

### Архитектура

использует архитектуру на основе COM, обеспечивая более гибкую и расширяемую структуру.

### Особенности

В первую очередь основан на Windows из-за зависимости от COM.

### Актуальность

OLE DB остается актуальным, но его использование в новых разработках не рекомендуется. OLE DB - это технология Microsoft, обеспечивающая универсальный доступ к данным из различных источников. Несмотря на то, что OLE DB до сих пор используется во многих приложениях, Microsoft рекомендует переходить на более современные технологии, такие как ADO.NET, для новых проектов.

### OraOLEDB

OLE DB Provider (OraOLEDB) – Это и есть провайдер , то есть ПО отвечающее за предоставление доступа к БД oracle.

### Подключение через ole db

## ODBC

Предназначен для унифицированного доступа приложений к **реаляционным** БД, использующих SQL.

**ODBC** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Open Database Connectivity*) — это программный интерфейс ([API](https://ru.wikipedia.org/wiki/API)) доступа к [базам данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), разработанный компанией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) в сотрудничестве с [Simba Technologies](https://ru.wikipedia.org/w/index.php?title=Simba_Technologies&action=edit&redlink=1) на основе спецификаций [Call Level Interface](https://ru.wikipedia.org/wiki/Call_Level_Interface) (CLI), который разрабатывался[[1]](https://ru.wikipedia.org/wiki/ODBC#cite_note-1) организациями [SQL Access Group](https://ru.wikipedia.org/wiki/SQL_Access_Group), [X/Open](https://ru.wikipedia.org/wiki/X/Open) и [Microsoft](https://ru.wikipedia.org/wiki/Microsoft). Впоследствии CLI был стандартизован [ISO](https://ru.wikipedia.org/wiki/ISO)[[2]](https://ru.wikipedia.org/wiki/ODBC#cite_note-2). Стандарт CLI призван унифицировать программное взаимодействие с [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%A3%D0%91%D0%94), сделать его независимым от поставщика СУБД и программно-аппаратной платформы.

### Архитектура

использует архитектуру на основе драйверов, где для каждого типа базы данных требуются определенные драйверы.

### Актуальность

ODBC (Open Database Connectivity) в настоящее время не является устаревшей технологией, но его актуальность несколько снизилась по сравнению с пиком популярности. Современные технологии, такие как ORM (Object-Relational Mapping) и более новые стандарты подключения к базам данных, часто используются вместо ODBC, особенно в новых проектах. Однако ODBC по-прежнему широко используется и остается востребованным для обеспечения совместимости с устаревшими системами и в тех случаях, когда требуется низкоуровневый доступ к базе данных.

## ORM

ORM (Object-Relational Mapping)

## ADOBD / ADO.NET

<https://stackoverflow.com/questions/3766433/what-is-difference-in-adodb-and-oledb>

Adodb (ActiveX Data Objects DB) — это API-слой поверх OLE DB. Он хорошо работает с базами данных на базе MS, такими как Sql Server, предоставляя согласованный API и оптимизации. При этом вы можете использовать ADODB для подключения к источникам данных, отличным от MS, но это будет означать, что вам потребуется поставщик OLEDB/ODBC для источника данных.

Проще говоря, для подключения к любому источнику данных вам нужен драйвер. Вот несколько распространенных сценариев, которые стоит рассмотреть:

ADODB для источника данных, имеющего только драйвер ODBC — ADODB использует поставщика OLEDB для ODBC, который загружает драйвер ODBC, который затем подключается к источнику данных.

ADODB для источника данных с драйвером OLEDB (например, SQL Server) — ADODB использует поставщика OLEDB для SQL Server для прямого взаимодействия с базой данных.

Oledb (Object Linking and Embedding DB) — это стандартный формат, поддерживаемый большим количеством баз данных, поэтому вы можете подключаться к Oracle, DB2 и т. д. с помощью Oledb. Вы также можете использовать OLEDB напрямую для подключения к Sql Server, но API более запутан по сравнению с подключением adodb, которое оптимизировано для работы с Sql Server и MS Access.

ADO.Net — это «архитектура» подключения к базе данных на основе .Net. В ADO.Net есть библиотека для Oledb — System.Data.OledbClient. Adodb был заменен/обновлен, и теперь ADO.Net использует библиотеку System.Data.SqlClient для баз данных/поставщиков данных на основе MS.

## ODP.NET

ODP.NET — поставщик данных Oracle для доступа к Oracle Database из приложений . NET

ODAC — это Oracle Data Access Components (одним из которых является ODP.NET) . Сюда входят другие компоненты доступа к данным, такие как OO4O, OLEDB и т. д.

ODP.NET, или Oracle Data Provider for .NET, это компонент, предоставляемый Oracle, который позволяет .NET-приложениям взаимодействовать с базами данных Oracle, используя ADO.NET.

# Клиенты

## ODC

Oracle Database Client

Полноценный клиентский пакет ПО для подключения к Oracle, включающий все возможные драйверы и утилиты:

OCI (Oracle Call Interface)

OLE DB Provider (OraOLEDB)

ODBC Driver

ODP.NET (для .NET Framework)

SQL\*Plus, Oracle Utilities (expdp, impdp и др.)

Поддержка TNS (файлы tnsnames.ora, sqlnet.ora)

Устанавливают когда нужны все возможности Oracle (администрирование, утилиты), работа с старыми приложениями, требующими полного клиента.

## ODAC

Docs: <https://docs.oracle.com/en/database/oracle/oracle-data-access-components/index.html>

Урезанный клиент специально для разработки и подключения.

ODAC — это Oracle Data Access Components (одним из которых является ODP.NET) . Сюда входят другие компоненты доступа к данным, такие как OO4O, OLEDB и т. д.

Oracle Data Access Components (ODAC) - комплект библиотек и драйверов для доступа к Oracle из разных языков программирования (C#, Java, Python и др.):

ODP.NET (Oracle Data Provider for .NET) – для .NET Framework и .NET Core

Oracle Data Provider for .NET Core (кросс-платформенная версия)

Oracle OLE DB Provider (OraOLEDB)

ODBC Driver

Oracle Instant Client (минимальный клиент для OCI-подключений)

Утилиты (SQL\*Plus, TNSPING)

ODAC Xcopy Packages

<https://www.oracle.com/database/technologies/net-downloads.html>

Use XCopy Oracle Data Access Components (ODAC) packages to deploy ODAC to large numbers of machines. XCopy allows complete installation and configuration customization. ODAC XCopy packages include:

Oracle Data Provider for .NET

Oracle Providers for ASP.NET

Oracle Provider for OLE DB

Oracle OLAP Provider for OLE DB

Oracle Services for Microsoft Transaction Server

Oracle Instant Client

Когда использовать?

Разработка на .NET (Framework/Core)  
Подключение из Python, Java, Node.js (через Instant Client)  
Нужны все возможные драйверы в одном пакете

Выбирайте ODAC, если:

* Нужны все драйверы (ODP.NET, OLE DB, ODBC, Instant Client).
* Работаете с .NET Core, Python, Java (не только Microsoft-инструментами).
* Требуется SQL\*Plus и другие утилиты.

ODAC можно использовать вместо

## OCMT

Немного урезанный установщик ПО по сравнению с OCD.

Oracle Client for Microsoft Tools (OCMT) - Установщик программного обеспечения, необходимого для подключения к Oracle DB в Excel, Power BI и .NET приложений.

1. OCMT автоматически устанавливает:

Oracle Data Provider for .NET (ODP.NET) — для работы с .NET-приложениями.

OLE DB Driver (OraOLEDB) — для Power BI, Excel, SSIS.

ODBC Driver — для совместимости с различными инструментами.

Базовые утилиты (например, SQL\*Plus) — для администрирования.

2. Оптимизация для Microsoft-экосистемы

OCMT настраивает:

Реестр Windows (чтобы Power BI/Excel видели Oracle OLE DB провайдер).

Пути к библиотекам (чтобы .NET-приложения находили ODP.NET).

Сетевые параметры (например, TNS\_ADMIN для файла tnsnames.ora).

### Детали

Современные версии OCMT подключают приложения к oracle DB используя **ODP.NET.**

## OIC

Oracle Instant Client (OIC)

Сильно урезанный пакет ПО для подключения к oracle DB, и нужен только для совершения запросов к БД и получения ответов и всё (для программирования хватает).

Client Oracle Instant — это базовый облегченный клиент, который можно распаковать в место без установки, он содержит только уровень связи для подключения к Oracle

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# OLE DB

## Установка OLE DB

мануальная установка через msi для odac не поддерживается более

поэтому нужно устанавливать вручную Xcopy:

1. скачать Xcopy ODAC

[Oracle .NET, Visual Studio, and VS Code ODAC Downloads for Oracle Database](https://www.oracle.com/database/technologies/net-downloads.html)

1. разархивировать куда то
2. В readme есть инструкция
3. запустить через cmd (администратор) install.bat:

install.bat odp.net4 c:\oracle myhome true true c:\credentials

1. добавить папки для oledb и bin в oledb в Path.

# 

# Миграция данных

Разные инструменты для разных видов миграций данных из Oracle DB.

[Oracle Database Migration Technologies | Oracle](https://www.oracle.com/database/cloud-migration/technologies/)

## Инструменты Oracle

### Oracle DB -> Oracle DB

#### Data pump

Этот инструмент работает только с образом FREE, в лайт нет expdp/impdp

1) Под обычным пользователем создать таблицу и др данные

2) Под админом дать права этому самому пользователю на возможность создавать файлы в директории DATA\_PUMP\_DIR (оракл сам генерирует эти глобальные элиасы):

GRANT READ, WRITE ON DIRECTORY DATA\_PUMP\_DIR TO excel\_user;

можно под админом посмотреть какой путь у DATA\_PUMP\_DIR:

SELECT directory\_name, directory\_path FROM dba\_directories WHERE directory\_name = 'DATA\_PUMP\_DIR';

обычно это: /opt/oracle/admin/FREE/dpdump/3387218F0CE207F9E0630800580A0B6F/

3) Под тем обычным пользователем провести экспорт всех схем и таблиц, относящихся к этому пользователю:

expdp excel\_user/Excel123@FREEPDB1 \

schemas=excel\_user \

dumpfile=excel\_user.dmp \

logfile=export\_excel\_user.log \

directory=DATA\_PUMP\_DIR

после этого в директории DATA\_PUMP\_DIR будет созданы файл excel\_user.dmp, а также лог

4) перенести файл excel\_user.dmp и одной машины в другую машину, где поставлена пустая БД

Скопировать файл из контейнера на мой компьютер

docker cp <ID\_контейнера1>:DATA\_PUMP\_DIR/excel\_user.dmp .

docker cp 57b9369e48566de0ca509d50dbb532428d822513415a24c51d1200834df385ef:/opt/oracle/admin/FREE/dpdump/3387218F0CE207F9E0630800580A0B6F/excel\_user.dmp .

из моего компьютера перенести файл в другой контейнер

docker cp excel\_user.dmp <ID\_контейнера2>:DATA\_PUMP\_DIR

docker cp excel\_user.dmp 4a326ee451a91c910f97734f430baea23b8552e958a0b4ba53337cf8d9ae2dd4:/opt/oracle/admin/FREE/dpdump/3387218F0CE207F9E0630800580A0B6F/

5) Выполнить импорт в другую БД

Заходим в другой контейнер с пустой БД

Под админом создаём такого же пользователя и выдаём ему права если надо:

sqlplus system/MyPass123@localhost:1521/FREEPDB1

CREATE USER excel\_user IDENTIFIED BY Excel123;

GRANT CONNECT, RESOURCE TO excel\_user;

ALTER USER excel\_user QUOTA UNLIMITED ON USERS;

GRANT READ, WRITE ON DIRECTORY DATA\_PUMP\_DIR TO excel\_user;

выходим из админа

Выполняем импорт из файла предыдущей БД:

impdp excel\_user/Excel123@FREEPDB1 \ schemas=excel\_user \ dumpfile=excel\_user.dmp \ logfile=import\_excel\_user.log \ directory=DATA\_PUMP\_DIR