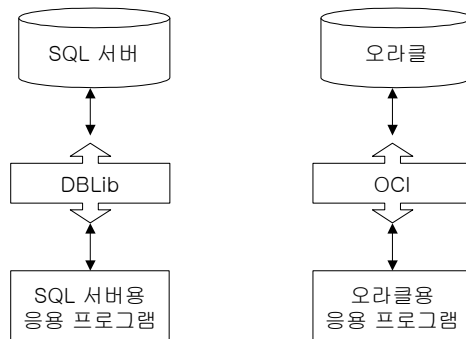


ADO.Net 1

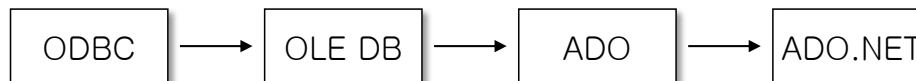


ADO.NET

- 닷넷은 데이터베이스 클라이언트 제작 언어로 충분한 기능을 제공한다.
 - DB 접속 라이브러리
 - 마법사와 코드 생성툴
 - DB 설계와 테이블 관리툴
- DBMS와 클라이언트가 통신하려면 접속 라이브러리가 필요하다.



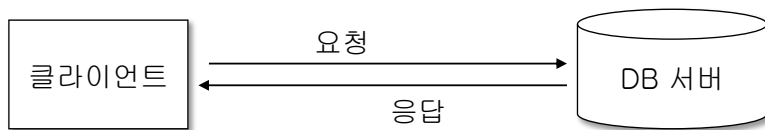
- 전용 라이브러리는 호환성이 부족하다.
- 범용 접속 인터페이스가 필요해졌다. ADO.NET은 그 중의 하나이다.



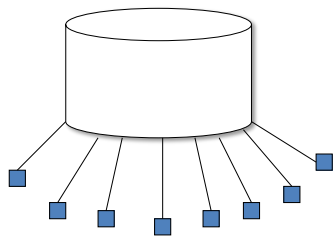


특징

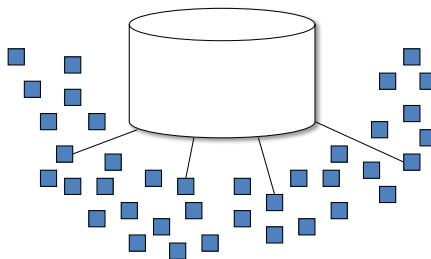
- ADO는 서버와의 연결을 항상 유지한 채로 동작하는 연결형이다.
- 현재성은 높지만 서버 부담이 높고 라이선스 비용이 늘어난다.



- ADO.NET은 서버와의 연결이 끊어진 채로 동작하는 비연결형을 지원한다. 필요할 때만 연결하여 서버의 부담을 덜어준다.



연결형 : 항상 연결되어 있다.



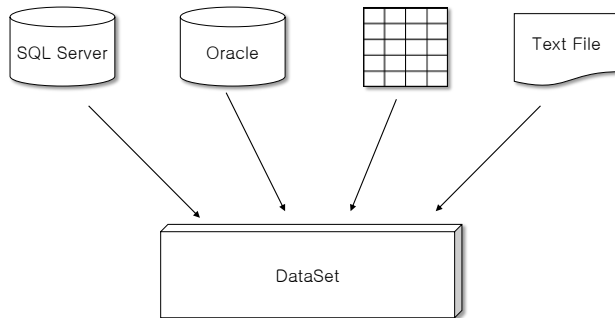
비연결형 : 필요할 때만 연결한다.

- 서버는 데이터 저장 역할만 하면 되니 서버에 부담이 적어 더 많은 동시 사용자를 지원할 수 있다.



DataSet

- DataSet은 관계형 데이터베이스 하나에 대응되는 오프라인 DB이며 연결이 끊어진 동안 서버의 역할을 대신한다.
- 이기종의 데이터나 비관계형 데이터까지 포함한다.



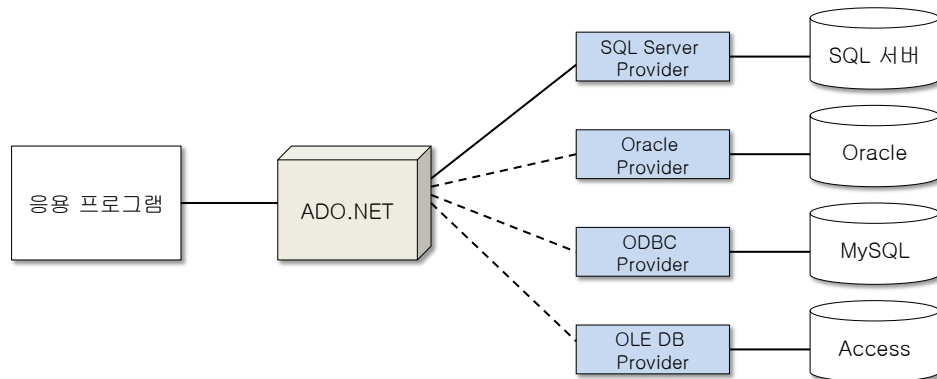
특성	연결형	비연결형
메모리 소모량	적다	많다
기동속도	빠르다	느리다
현재성	높다	낮다
새로운 개체 생성	가능	불가능
서버의 부담	높다	낮다
액세스 속도	느리다	빠르다
타입 체크	불가능	엄격하다

- 정적인 사본을 사용하므로 쉽고, 오류 가능성이 낮으며 빠르다.
- 리소스를 많이 소모하며 현재성이 떨어지는 단점이 있다.
- 닷넷은 XML과 SOAP을 통한 데이터 교환을 지원하여 상호 운용성이 높다.



구조

- 공급자(Provider) : DB에 연결하고 명령을 실행하는 모듈로서 응용 프로그램에게 DBMS의 데이터를 공급한다.



- 인터페이스를 통일해 DBMS를 일관된 방법으로 프로그래밍할 수 있다.
- 기본 제공 공급자는 다음과 같다. MySQL, DB2, PostgreSQL 등에 대한 공급자를 추가 설치할 수 있다.

공급자	설명	네임스페이스
SQL Server 공급자	SQL 서버 7.0 이상의 버전을 액세스한다.	System.Data.SqlClient
Oracle 공급자	오라클 8.1.7 이상의 버전을 액세스한다.	System.Data.OracleClient
OLE DB 공급자	OLE DB를 통해 DBMS에 연결한다.	System.Data.OleDb
ODBC 공급자	ODBC를 통해 DBMS에 연결한다.	System.Data.Odbc



공급자

- SQL 서버 공급자가 가장 성능이 좋다.
- 각 공급자는 4개의 주요 클래스를 제공한다.

클래스	기반 인터페이스	설명
Connection	IDbConnection	데이터 소스와 연결한다.
Command	IDbCommand	SQL 명령이나 저장 프로시저를 실행한다.
DataReader	IDbDataReader	읽기 전용, 전진 전용의 결과셋을 표현한다.
DataAdapter	IDbDataAdapter	데이터 원본을 읽거나 수정한다.

- 클래스의 실제 이름은 DBMS 접두가 붙어 공급자마다 조금씩 다르다.
SqlConnection, OleDbConnection 등
- 동일한 인터페이스로부터 상속받아 멤버의 구성이나 기능은 거의 동일하다.
- 하나만 잘 배워두면 다른 공급자도 같은 방식으로 사용할 수 있다.



샘플 데이터베이스

- SQL 서버(2019 이후 버전) 설치후 ADOTest DB를 생성하고 테이블을 생성한다.

```
CREATE TABLE tblPeople
(
    Name NVARCHAR(10) PRIMARY KEY,
    Age INT NOT NULL,
    Male BIT NOT NULL
);

INSERT INTO tblPeople VALUES ('정우성', 36, 1);
INSERT INTO tblPeople VALUES ('고소영', 32, 0);
INSERT INTO tblPeople VALUES ('배용준', 37, 1);
INSERT INTO tblPeople VALUES ('김태희', 29, 0);
```

```
CREATE TABLE tblSale
(
    OrderNo INT IDENTITY (1,1) PRIMARY KEY,
    Customer NVARCHAR(10) NOT NULL REFERENCES tblPeople(Name),
    Item NVARCHAR(20) NOT NULL,
    ODate DATETIME NOT NULL,
);

INSERT INTO tblSale (Customer, Item, ODate) VALUES ('정우성', '면도기', '2008/1/1');
INSERT INTO tblSale (Customer, Item, ODate) VALUES ('고소영', '화장품', '2008/1/2');
INSERT INTO tblSale (Customer, Item, ODate) VALUES ('김태희', '핸드폰', '2008/1/3');
INSERT INTO tblSale (Customer, Item, ODate) VALUES ('김태희', '휘발유', '2008/1/4');
```

- 회원과 쇼핑 내역이며 tblPeople의 Name 필드와 tblSale의 Customer는 외래키 관계로 연결되어 있다.

	Name	Age	Male
1	고소영	32	0
2	김태희	29	0
3	배용준	37	1
4	정우성	36	1

	OrderNo	Customer	Item	ODate
1	1	정우성	면도기	2008-01-01 00:00:00,000
2	2	고소영	화장품	2008-01-02 00:00:00,000
3	3	김태희	핸드폰	2008-01-03 00:00:00,000
4	4	김태희	휘발유	2008-01-04 00:00:00,000

- 참고 : 위 스크립트를 이해하지 못하면 아직 ADO.NET을 배울 준비가 안된 것이다. SQL부터 먼저 공부하고 오자.