

Overview of PL/SQL



1. PL/SQL 개념

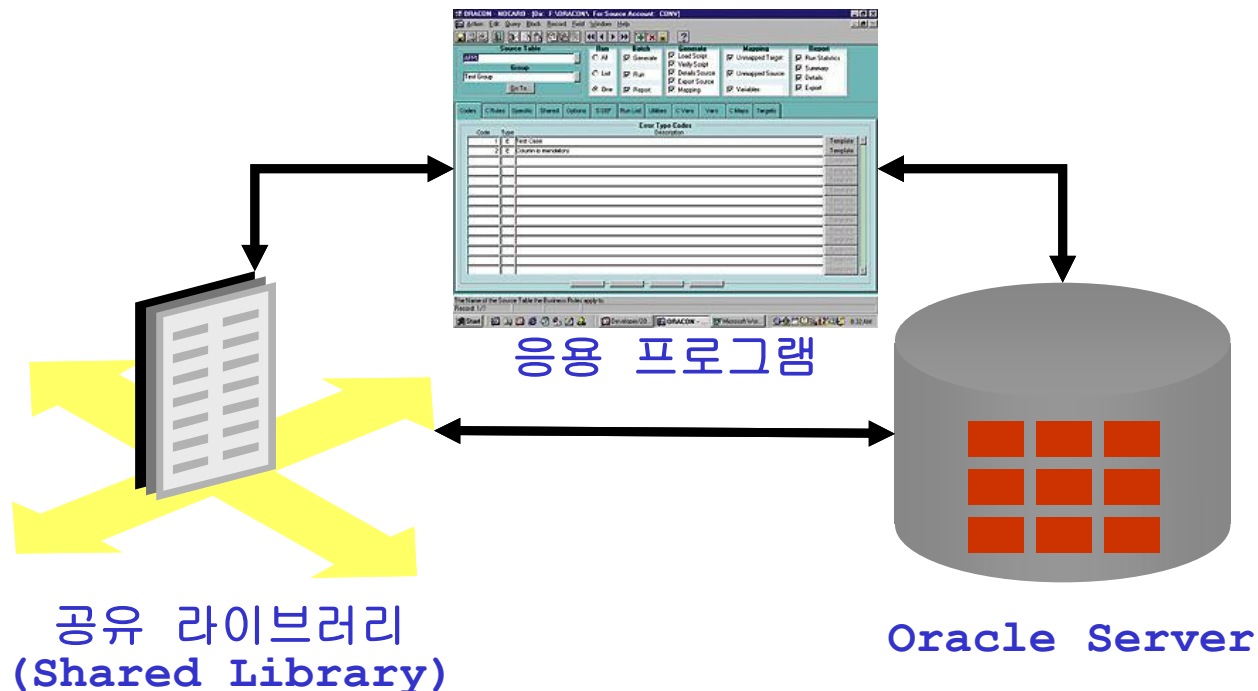
가
DB
PL/SQL DB (*)

- ❑ **Procedural Language/SQL (PL/SQL)**은 **SQL**에 Oracle Corporation의 절차적 언어를 추가한 것으로 **관계형 데이터베이스의 표준 데이터 액세스 언어**
- ❑ 1970년대와 1980년대에 설계된 **프로그래밍 언어**의 다양한 고급 기능 포함
- ❑ SQL의 데이터 조작문 (DML) 및 질의 (Query) 문을 블록 구조 및 프로시저 단위 코드에 포함시켜 더욱 강력한 트랜잭션 처리 제공

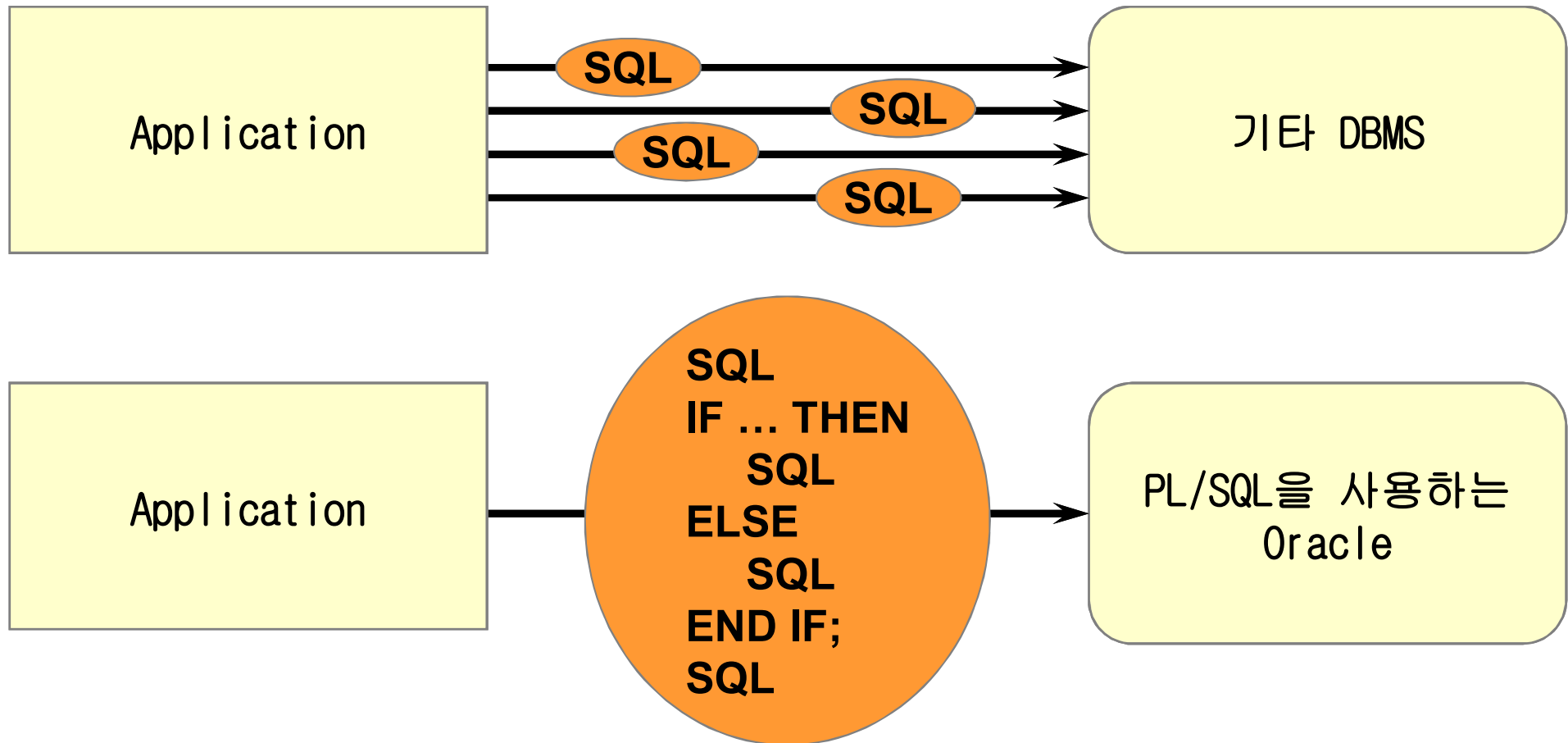
2. PL/SQL 장점

2-1. 통합 (Integration)

- PL/SQL은 Oracle Server 및 Oracle 개발 도구 모두에서 중심 역할
- SQL 데이터 유형은 PL/SQL에서도 사용 가능
- Oracle Developer와 같은 Oracle 도구에는 Oracle Server에 있는 엔진과는 별도의 PL/SQL 엔진이 있음



2-2. 성능 향상



Handwritten signature

3. PL/SQL 블록 구조

3-1. PL/SQL 블록 구조

DECLARE (선택항목)

가

변수, 커서, 사용자가 정의한 예외 사항

BEGIN (필수항목)

- SQL 문
- PL/SQL 문

EXCEPTION (선택항목)

오류가 발생할 때의 수행 작업

END; (필수항목)

DECLARE



BEGIN



EXCEPTION



END;

3-2. SQL*Plus에서 명령문과 PL/SQL 블록 실행

- SQL 문 또는 PL/SQL 제어문 끝에 **세미콜론 (;)** 입력
- DECLARE, BEGIN, EXCEPTION 등의 섹션 키워드 뒤에는 세미콜론을 사용하지 않지만 **END 및 기타 모든 PL/SQL 문에는 세미콜론 사용**
- 슬래시 (/)를 사용하여 SQL*Plus 버퍼에서 익명 (anonymous) PL/SQL 블록 실행
- 블록이 성공적으로 실행되면 다음 메시지 출력

PL/SQL procedure successfully completed.

- 버퍼에서 PL/SQL 블록은 하나의 연속적인 명령문으로 취급

4. 블록 유형

Anonymous (X)

```
[DECLARE]

BEGIN
    --statements

[EXCEPTION]

END ;
```

Procedure SQL DML

```
PROCEDURE name
IS

BEGIN
    --statements

[EXCEPTION]

END ;
```

Function SQL 가

```
FUNCTION name
RETURN datatype
IS

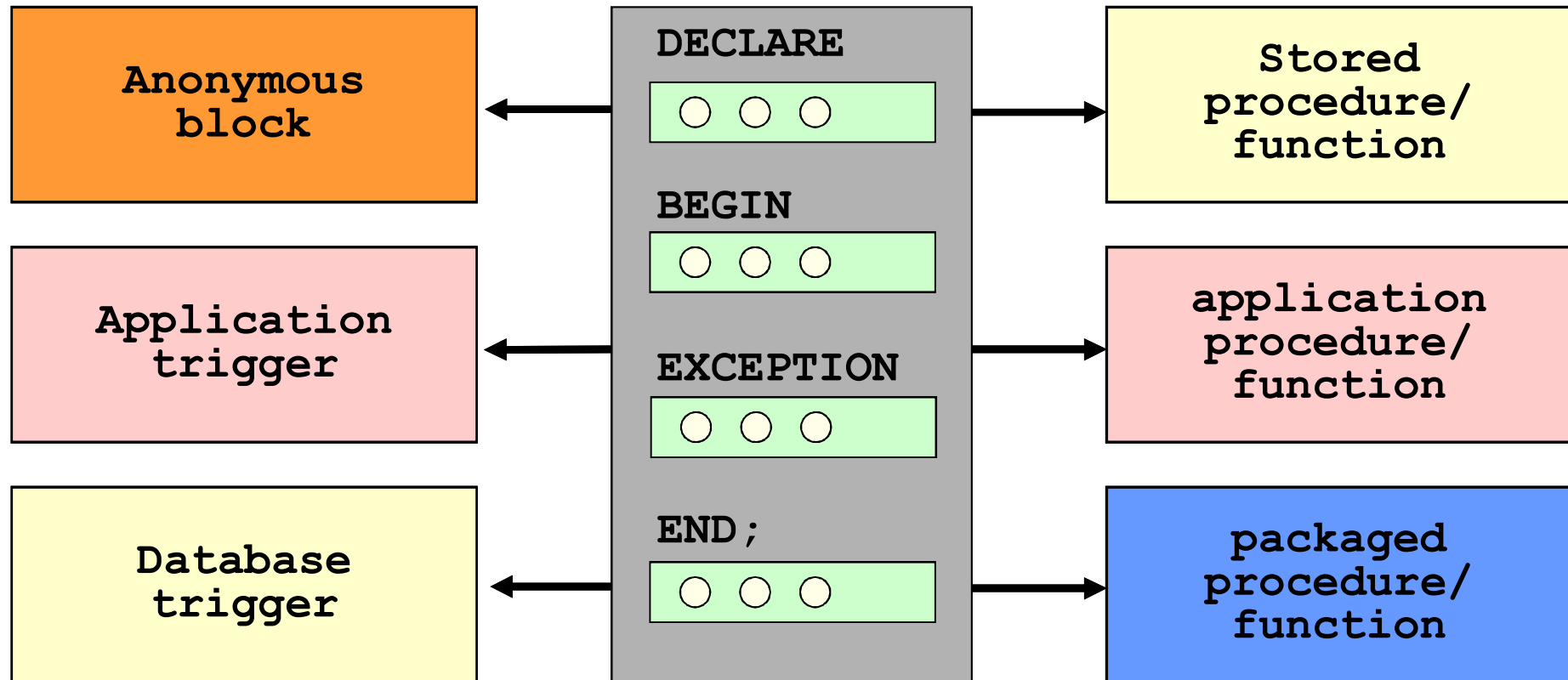
BEGIN
    --statements
    RETURN value;

[EXCEPTION]

END ;
```

- PL/SQL의 모든 단위는 하나 이상의 블록으로 구성
- 블록은 완전히 분리되거나 다른 블록 안에 중첩 가능

5. 프로그램 생성자



- 모든 PL/SQL 환경
- 오라클 서버
- Oracle Develop
- 오라클 서버 / Oracle Develop

mbg