

## 72. Client Load Balancing

서비스를 사용하는 이유

=> RAC 를 사용하다보니 인터커넥트부하로 성능이 느린 단점이 발생해서 그 단점을 극복하고자 나온 기능이다. 업무 파티션을 해서 업무마다 노드를 할당 해주게 하는 기술적인 기능이 바로 서비스다.

=> 서비스 + 스케줄 -> 서비스 효과

서비스 + 리소스 매니저 -> 시너지 효과

=> 서비스 단위로 trace file을 생성할 수 있어서 특정 서비스가 왜 느린지 더 큰 단위로 분석을 할 수 있게 되었다.

RAC의 장점인 고가용성에 대한 이론

1. Client Load Balancing
2. Client CTF(connect time failover)
3. 서버측 load balancing
4. TAF(transparent application failover)
5. FAN(fast application notification)

RAC 환경에서 제공하는 load balancing 방법 2가지

1. Client Load Balancing: 클라이언트 (원도우)의 tnsnames.ora 파일 안에 load\_balance=on 이라는 파라미터가 있어서 어느 노드로 접속할지가 완전히 랜덤으로 하는 load balancing

2. Server Load Balancing: 서버의 부하에 따라 부하가 적은 노드로 접속되게 하는 load balancing

문제 1. 1번 노드에 tnsnames.ora에 racdb 기본 서비스를 이용하는 tns 정보를 이용해서 scott으로 접속하시오.

cd \$ORACLE\_HOME/network/admin

=> racdb\_taf 있는지 확인하기

```
racdb_taf=
(DESCRIPTION =
  (address_list=
    (load_balance=on)
    (failover=on)
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.56.111)(PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.56.112)(PORT = 1521)) )
  (CONNECT_DATA =
    (SERVICE_NAME = racdb)
    (failover_mode=(type=select)(method=basic) )
  )
)
```

sqlplus scott/tiger@racdb\_taf

select instance\_name from v\$instance;

2번 노드의 listener 올리기

(2번 노드)

lsnrctl start

=> 1번 노드와 2번 노드 둘 다 나오는지 확인하기