

Banco de Dados II

O Oracle Distribuído
Links de BDs
Transparência de Localização

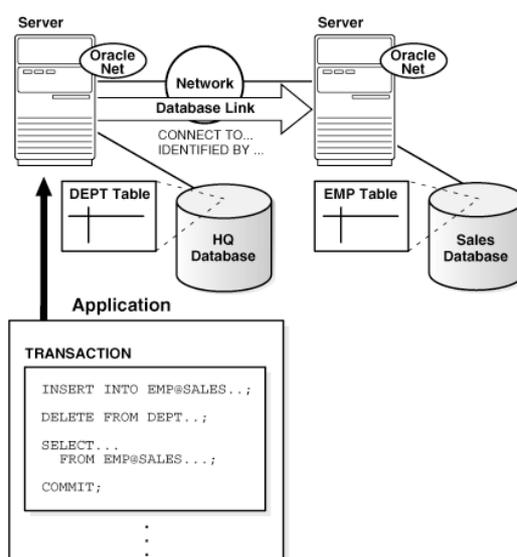
BDD Oracle

- Clientes e Servidores
 - **Servidor**: software Oracle que gerencia um banco de dados
 - **Cliente**: aplicação que solicita informação para um servidor.
 - Cada computador no sistema BDD é um **nó**.
 - Um nó age como um cliente, um servidor, ou ambos dependendo da situação
- Cada BD tem o seu próprio nome do banco de dados global dentro da rede e do SBDD

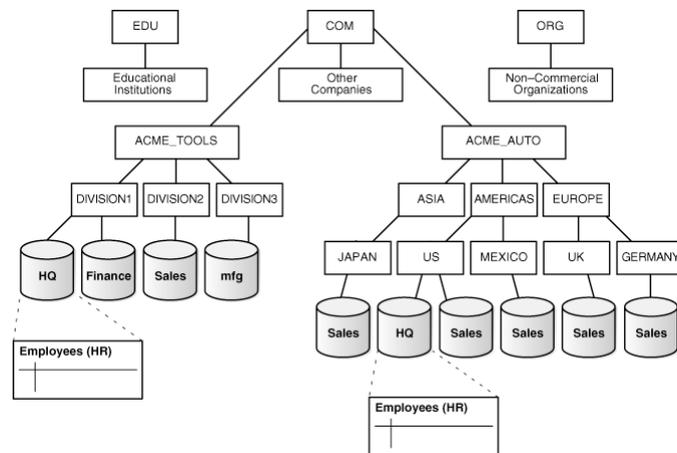
BDD Oracle

- **Oracle Net**
 - Serviço Oracle utilizado para facilitar a comunicação através da rede
 - Permite a comunicação e suporte a transações remotas e distribuídas
- **O Oracle Net deixa transparente a conectividade que é necessária para transmitir requisições SQL e receber dados de aplicações que usem o sistema**

Oracle Distribuido



Nomes de BD Globais



Exemplos:

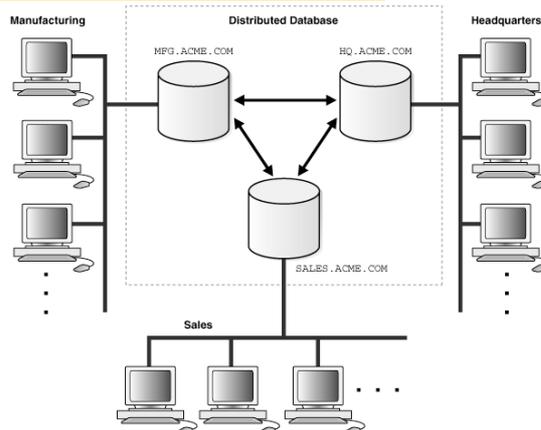
hq.division1.acme_tools.com

sales.us.americas.acme_auto.com

Oracle homogêneo

- Um sistema de bds distribuídos Oracle pode incorporar bds Oracle de diferentes versões

- Parece um acesso local a dados



Oracle Heterogêneo

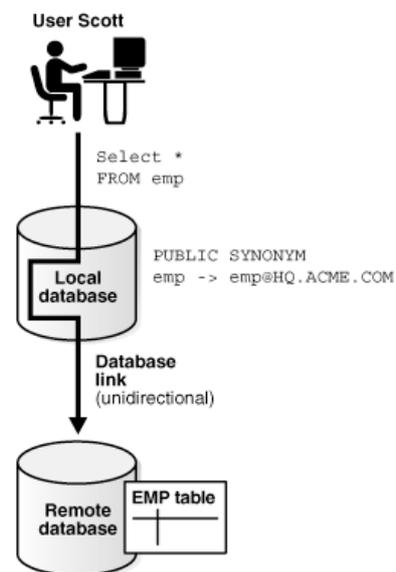
- Neste caso, pelo menos um dos bds envolvidos não é Oracle
- Para a aplicação, todo o sistema distribuído funciona como um único e local bd Oracle
- O acesso do bd Oracle ao bd nao-Oracle é feito de duas formas:
 - Usando os serviços heterogêneos em conjunto com um agente (o Gateway transparente)
 - Usando a conectividade genérica (desde que os bds nao-Oracle suportem os protocolos ODBC e OLE DB)

Links de BDDs

- Conceito central em sistemas de bds distribuídos
- Funciona como uma conexão entre dois bds físicos permitindo a um cliente acessá-los como um único bd lógico
 - Um usuário local pode acessar um BD remoto sem ter uma conta (usuário) naquele BD
- É uma comunicação “one-way”

Links de BDD - Oracle

- É armazenado como entrada no dicionário de dados ou catálogo
- Para a conexão ocorrer, cada bd do sistema distribuído precisa ter um nome global único no domínio de rede



Segurança com Links

- Públicos (opção PUBLIC)
 - Todos os usuários e todos os subprogramas PL/SQL daquele servidor de bd podem usá-lo
- Privados (padrão)
 - Apenas quem criou o link e os subprogramas PL/SQL daquele esquema podem usá-lo

Compartilhamento de Links

- **Compartilhados (opção SHARED)**
 - Múltiplas transações cliente podem usar o mesmo link simultaneamente
- **Dedicados (padrão)**
 - Apenas a transação cliente em uso pode usar o link em questão

Tipos de Links

Tipo	Descrição
Link de Usuário Conectado	Usuários se conectam como eles mesmos Eles devem ter uma conta no BD remoto com o mesmo nome de usuário e senha do BD local
Link de Usuário Fixo	Usuários se conectam usando um nome de usuário e senha referenciado no link
Link de Usuário Corrente	Um usuário se conecta como um usuário global

Criando Links de BDs

- Usa-se a declaração **CREATE DATABASE LINK**
 - Tem de ter o **privilégio de sistema** CREATE DATABASE LINK
 - Tem de ter o **privilégio de sistema** CREATE SESSION no BD Oracle remoto
 - Oracle Net deve estar instalado em ambos BD Oracle local e remoto
- Após a criação é possível especificar os objetos do esquema em declarações SQL

Criando Links de BDs

- **Sintaxe:**
 - CREATE [SHARED][PUBLIC] DATABASE LINK *link_name* [CONNECT TO *user* IDENTIFIED BY *password*] [AUTHENTICATED BY *user* IDENTIFIED BY *password*] [USING '*connect_string*']
 - A **connect string** precisa estar definida no Net, no arquivo de configurações de nome TNSNAMES.ORA.
- **Exemplo:**
 - CREATE DATABASE LINK test
CONNECT TO jje IDENTIFIED BY jje USING 'test';

Criando Links de BDs

- Sintaxe:
 - CREATE [SHARED][PUBLIC] DATABASE LINK *link_name* [CONNECT TO CURRENT_USER] [USING '*connect_string*']

- Exemplo:

```
CREATE DATABASE LINK
sales.us.americas.acme_auto.com Connect to
current user USING 'sales_us';
```

Exemplos de Criação

- CREATE PUBLIC DATABASE LINK foo CONNECT TO CURRENT_USER USING 'am_sls';
- CREATE DATABASE LINK sales.us.americas.acme_auto.com CONNECT TO scott IDENTIFIED BY tiger USING 'sales_us';
- CREATE PUBLIC DATABASE LINK sales CONNECT TO scott IDENTIFIED BY tiger USING 'rev';

Para Criar Link de BDs

127.0.0.1:8080/apex/f?p=4500:1001:2336162365629230::CREATE:NO:OB_CURRENT_TYPE:DATABASE_LINK

ORACLE Database Express Edition

Usuário: DAM111

Início > [Browser de Objetos](#)

Link De banco de dados

Definir
Confirmar

Criar Link de Banco de Dados Cancelar Próximo >

Um link de banco de dados é um objeto de esquema em um banco de dados que permite acessar objetos contidos em outro banco de dados. Após criar o link de banco de dados, você pode acessar os objetos remotos acrescentando @**dblink** ao nome da tabela ou da view, onde dblink é o nome do link especificado nesta página.

Esquema: DAM111

- Nome de Link do Banco de Dados
- Conectar Ao Esquema
- Senha
- Nome de Host ou IP Remoto
- Porta de Host Remota 1521
- SID ou Nome do Serviço Nome do Serviço SID

Exemplo de código

```

Create database link TESTE
connect to ttt identified by ttt
using '(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)
      (HOST = 10.0.41.182)(PORT = 1521))
  )
(CONNECT_DATA = (SID = XE)
)
)'
```

Para ver privilégios

```
SELECT DISTINCT PRIVILEGE AS "Database  
Link Privileges"  
FROM ROLE_SYS_PRIVS  
WHERE PRIVILEGE IN  
  ('CREATE SESSION','CREATE DATABASE  
  LINK', 'CREATE PUBLIC DATABASE LINK');
```

Acesso aos Objetos via Link

- `schema.schema_object@global_database_name`
- Exemplo:
 - `SELECT * FROM hr.emp@hq.acme.com;`
 - `SELECT * FROM
scott.emp@sales.division3.acme.com;`

Outros Comandos

- Para testar:
 - `SELECT * FROM emp@foo;`
 - Tem de ter autorização no BD remoto para acessar objetos específicos
- `DROP [PUBLIC] DATABASE LINK link_name`
- `ALTER SESSION CLOSE DATABASE LINK link_name`
- `SELECT DB_LINK FROM USER_DB_LINKS;`

Restrições

- Não pode:
 - Conceder privilégios sobre objetos remotos
 - Executar describe sobre objetos remotos, exceto tabelas, views, procedures e functions
 - Definir integridade referencial
 - Conceder roles para usuários no BD remoto

Transparência de Localização

- Pode-se fazer a funcionalidade do SBDD transparente para usuários que trabalham com o sistema
 - O objetivo da transparência é fazer um sistema de banco de dados distribuídos aparecer como um simples banco de dados
- *Transparência local* existe quando um usuário pode referir-se a um objeto no banco de dados sem se preocupar com o nó onde se encontra este objeto

Transparência de Localização

- Sinônimos
 - Escondem a “*identidade*” de um objeto, incluindo sua localização no BDD
- Views
 - Pode-se criar *views locais* que façam joins entre tabelas locais e remotas
- Procedures
 - Pode-se usar procedures que *referenciem dados remotos*
 - Pode-se “chamar” procedures remotas

Sinônimos

- Os sinônimos podem conferir transparência na localização dos objetos, pois escondem a identidade dos mesmos
 - Se o objeto referido pelo sinônimo mudar para outro local, só é necessário alterar a definição do sinônimo
- Os sinônimos devem ser públicos e têm que ser declarados em todos os BDs locais

Exemplo

- Suponha a consulta:

```
SELECT * FROM emp@hq.acme.com;
```
- Pode-se criar um sinônimo emp para emp@hq.acme.com
- Assim, a consulta é reescrita para:

```
SELECT * FROM emp;
```

Sinônimos para

- Tables
- Types
- Views
- Materialized views
- Sequences
- Procedures
- Functions
- Packages

Sintaxe

CREATE or replace [PUBLIC] synonym_name FOR
[schema.]object_name[@database_link_name];

- Onde:
 - **PUBLIC** – sinônimo disponível para todos os usuários
 - Se omitido, o sinônimo fica privado (usado pelo seu criador)
 - Tem de ter o privilégio de sistema CREATE PUBLIC SYNONYM
 - **Schema**: esquema do objeto; se omitido, será o esquema do criador do sinônimo
 - **Object_name**: tabela, view, sequence, etc.
 - **Database_link_name**: nome do link do BD remoto

Outro exemplo

- `Select * from dono1.cliente;`
- `Create or replace public synonym cliente for dono1.cliente;`
- Vantagens:
 - Ao acessar, não preciso colocar o dono do objeto
 - Os usuários não irão saber quem é o dono (segurança)
 - Simplificação da codificação
 - Qualquer usuário do banco pode executar o select

** tem de ter privilégio sobre o objeto destino

Observações

- Um sinônimo é uma referência a um objeto.
 - Um usuário que tem acesso a um sinônimo de um objeto de esquema particular, deve ter privilégios também sobre o objeto em si.
- Somente usuários com Privilegio DBA podem criar Sinônimos Públicos
- Para remover:
 - `DROP SYNONYM <synonym_name>`
 - ou
 - `DROP PUBLIC SYNONYM <synonym_name>;`

Views

- Podem ser usadas para criar transparência local para tabelas locais e remotas no bd distribuido
- Exemplo – considere:

- Tabela emp no bd local
- Tabela depto no bd remoto

```
CREATE VIEW Companhia AS
SELECT e.empnum, e.nome, d.nomedepo
FROM bda.emp e, hrr.depto@hq.acme.com d
WHERE e.deptnum = d.deptnum;

SELECT * FROM companhia;
```

Procedures

- Podem também prover transparência de localização:
 - Procedures locais podem referenciar dados remotos
 - Pode-se usar procedures locais para chamar procedures remotas
 - Pode-se usar sinônimos para referenciar procedures remotas

Exemplo

- Procedure referenciando dados remotos

Create or replace procedure demite_emp (enum
number) as

Begin

Delete from emp@hq.acme.com

where empno = enum;

End;

Exemplo

- Procedure referenciando dados remotos
através de sinônimo

Create Synonym emp for emp@hq.acme.com;

Create or replace procedure demite_emp (enum
number) as

Begin

Delete from emp

Where empno = enum;

End;

Exemplo

- Chamando procedure remota
 - Connect `hrr@hq.acme.com`
Create procedure `term_emp` (enum number) as
Begin
 delete from emp where empno = enum;
End;
GRANT EXECUTE ON `term_emp` TO scott;
 - Connect `scott@local_db`
Create procedure `demite_emp` (enum number) as
Begin
 Execute `term_emp@hq.acme.com`;
End;

Observações

- Quando uma procedure local inclui um comando que referencia uma tabela ou view remota, o dono da procedure local pode conceder privilégios de execução para qualquer usuário
 - O usuário que recebe o privilégio pode executar a procedure e indiretamente acessar os dados remotos
 - **Cuidado:** privilégios de objetos referenciados dentro de uma procedure não precisam ser explicitamente concedidos para usuários que executam as mesmas

Observações

- Restrições em relação às sentenças DML e DDL em BDs remotos:
 - Updates em **objetos remotos não-oracle** não podem fazer referencia a objetos locais
 - Colunas LONG e LONG RAW, sequencias devem estar **no mesmo nó**
 - Não é possível executar comandos **DDL remotos** (ex. CREATE, ALTER, DROP) em sistemas homogêneos exceto através da execução de procedures do package DBMS_SQL package
 - Funções como **sysdate**, **user** são executadas com base nas informações do bd local