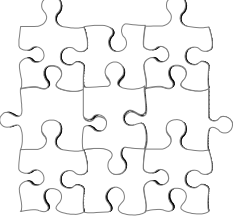


Объекты базы данных СУБД Oracle



Немногие назовут имена тех, кто осудил Галилея. Но каждый знает, в какой системе они работали.

Ю. Кусьмерек

Корпоративные системы баз данных
© М.Л. Цымблер

Содержание

- Понятие базы данных в СУБД Oracle
- Понятие схемы
- Объекты схемы
- Словарь данных

Корпоративные системы баз данных
© М.Л. Цымблер
2

База данных СУБД Oracle

- *База данных для СУБД Oracle – набор взаимосвязанных баз данных (база данных понимается в смысле классического определения Дейта).*
- *База данных может находиться в состоянии **online** (доступна для использования) или **offline** (закрыта администратором базы данных для выполнения административных операций).*
- *База данных имеет **логическую** и **физическую** структуру (концептуальный и физический уровни ANSI/SPARC архитектуры систем баз данных).*

Корпоративные системы баз данных
© М.Л. Цымблер
3

Схема данных

- *Схема (schema)* – это коллекция объектов логической структуры базы данных (таблицы, индексы, представления, хранимые процедуры и др.).
- Схема ассоциируется с именем пользователя-владельца ее объектов и имеет такое же имя. Пользователь имеет доступ ко всем объектам в своей схеме. Доступ пользователя к объектам "чужой" схемы возможен при наличии соответствующих привилегий.

Корпоративные системы баз данных

© М.Л. Цымблер

4

Пример: создание схемы

```
CREATE SCHEMA
  AUTHORIZATION painter
  CREATE TABLE pencil
    (color CHAR2(10) PRIMARY KEY,
    qty NUMBER)
  CREATE VIEW red_pencil AS
    SELECT color, qty FROM pencil
    WHERE color = 'RED'
  GRANT select ON red_pencil TO malevich
```

Корпоративные системы баз данных

© М.Л. Цымблер

5

Объекты схемы

- Таблицы
- Последовательности
- Представления
- Снимки
- Индексы
- Процедуры и функции
- Пакеты
- Синонимы
- Связи базы данных
- Триггеры
- Кластеры

Корпоративные системы баз данных

© М.Л. Цымблер

6

Таблицы

- *Таблица (table)* – основная единица хранения данных в базе данных. Для таблицы можно определить ограничения целостности и триггеры.

Корпоративные системы баз данных

© М.Л. Цымблер

7

Пример: создание таблицы

```
CREATE TABLE s (
  sid NUMBER
    CONSTRAINTS pk_sid PRIMARY KEY,
  sname CHAR(20)
    CONSTRAINTS nn_sname NOT NULL,
  scity CHAR(50)
    CONSTRAINTS nn_scity NOT NULL,
  address CHAR(100)
    CONSTRAINTS nn_address NOT NULL,
  rating NUMBER
    CONSTRAINTS ch_rating CHECK (rating>=0)
    DEFAULT 21)
```

Корпоративные системы баз данных

© М.Л. Цымблер

8

Последовательности

- *Последовательность (sequence)* генерирует уникальные порядковые номера, которые могут использоваться для реализации полей-счетчиков в одной или нескольких таблицах.
- Пример:

```
CREATE SEQUENCE inc_seq
  INCREMENT BY 1
```

Корпоративные системы баз данных

© М.Л. Цымблер

9

Представления

- *Представление (view)* – это виртуальная таблица, отображающая данные из одной или нескольких таблиц (*базовых таблиц*). Базовая таблица может быть представлением. Над представлениями разрешены те же действия, что и над таблицами.

```
CREATE VIEW Disp_S
(Name, City, Address, Rating) AS
SELECT SName, SCity, Address, Rating
FROM S
```

- Представление может быть обновлено
 - *явно* (запросом SQL), если оно удовлетворяет ряду ограничений (нет вычисляемых полей и др.);
 - *неявно* (с помощью триггеров – процедур, подменяющих собой запросы SQL на модификацию представления).

Снимки

- *Снимок (snapshot, materialized view)* – это копия таблицы или данных из нескольких таблиц.
- Снимки используются для резервного копирования базы данных и для ускорения обработки запросов в случае распределенных баз данных. Снимки подлежат периодическому обновлению.

Пример: создание снимка

```
CREATE SNAPSHOT sp_snap
PCTFREE 5 PCTUSED 60
TABLESPACE USERS
STORAGE
(INITIAL 50K NEXT 50K PCTINCREASE 50)
REFRESH FAST
START WITH sysdate
NEXT sysdate + 7
AS
SELECT * FROM sp@supply.chel_branch.ru
```

Индексы

- *Индекс (index)* – объект схемы для быстрого поиска данных в таблицах/представлениях.
- Пример:

```
CREATE INDEX S_city_idx ON S (SCity)
CREATE INDEX S_rating_idx ON S (Rating DESC)
```

Процедуры и функции

- *Процедура (procedure)* или *функция (function)* – это подпрограмма на языке *PL/SQL* (процедурном расширении языка *SQL*).
- Процедура и функция могут быть выполнены пользователем, имеющим соответствующую привилегию.

Пример процедуры

```
/* Пересчет рейтинга поставщиков.
recalc_date – дата пересчета */
CREATE PROCEDURE change_rating(recalc_date in date) AS
DECLARE
    f_sid NUMBER;
    f_rating NUMBER;
    cursor cur_s is
        select sid, rating from s;
BEGIN
    open cur_s;
    fetch cur_s into f_sid, f_rating;
    while cur_s%found
    loop
        f_rating = f_rating + sp_qty(f_sid, recalc_date)/100 –
            expire_qty(f_sid, recalc_date);
        insert into S values (f_rating);
        fetch cur_s into f_sid, f_rating;
    end loop;
    close cur_s;
END;
```

Пакеты

- Процедуры и функции могут быть объединены в *пакет (package)*. В пакете могут также храниться глобальные переменные и константы.
- Пакет имеет *интерфейсную секцию* (описание экспортируемых объектов) и *тело* (реализация экспортируемых объектов).

Корпоративные системы баз данных

© М.Л. Цымблер

16

Пример: создание пакета

```
-- Интерфейс пакета
CREATE PACKAGE dbms_pkg AS
    FUNCTION sp_qty(sid NUMBER, recalc_date DATE)
        RETURN NUMBER;
    FUNCTION expire_qty(sid NUMBER, recalc_date DATE)
        RETURN NUMBER;
    PROCEDURE change_rating(recalc_date DATE);
END dbms_pkg;
-- Тело пакета
CREATE PACKAGE BODY dbms_pkg AS
    -- Реализация подпрограмм пакета
END dbms_pkg;
```

Корпоративные системы баз данных

© М.Л. Цымблер

17

Синонимы

- *Синоним (synonym)* или *алиас (alias)* – дополнительное имя объекта, ссылка на объект. Синонимы используются для маскировки истинного имени объекта и/или обеспечения прозрачного доступа к объекту.
- Пример:

```
CREATE SYNONYM cr
    FOR change_rating
CREATE PUBLIC SYNONYM sp_data
    FOR SP
CREATE SYNONYM pencil
    FOR painter.pencil@paintmasters.louvre.fr;
```

Корпоративные системы баз данных

© М.Л. Цымблер

18

Связи базы данных

- *Связь базы данных (database link)* – объект в локальной базе данных, описывающий путь к удаленной базе данных.
- Пример:

```
CREATE DATABASE LINK paintmasters.louvre.fr
CONNECT TO malevich
IDENTIFIED BY red_square;
SELECT * FROM red_pencil@paintmasters.louvre.fr;
```

Триггеры

- *Триггер (trigger)* – особая хранимая процедура, состоящая из двух частей: событие триггера и тело триггера.
- *Событие триггера* – это добавление, удаление или обновление записи (-ей) конкретной таблицы или представления.
- *Тело триггера* – программный код на языке PL/SQL; может использовать вызовы хранимых процедур и функций, Java, C.
- При наступлении события триггера СУБД Oracle автоматически выполняет тело триггера; триггер не может быть выполнен явно пользователем.

Пример: триггер

```
-- Общий вес деталей в поставке не должен превышать 1,5 тонн.
CREATE TRIGGER check_weight
BEFORE INSERT OR UPDATE ON SP
FOR EACH ROW
DECLARE
  p_weight, total_weight NUMBER;
BEGIN
  select weight into p_weight from P where pid = :new.pid;
  total_weight = :new.qty * p_weight;
  if total_weight > 1500 then
    raise_application_error(-20601, 'Общий вес деталей в поставке
(Код_детали=' || TO_CHAR(:new.pid) || ', Код_поставщика=' ||
:new.sid || ') равен ' || TO_CHAR(total_weight) || ' кг.' ||
'Вес поставки не должен превышать 1500 кг!');
  end if;
END;
```

Кластеры

- *Кластер (cluster)* – объект схемы, задающий способ хранения таблиц. Таблицы, связанные посредством внешнего ключа, после их кластеризации физически будут храниться в одних и тех же блоках данных.
- Кластеры сокращают время доступа к данным на диске при выполнении операции JOIN по кластеризованным таблицам.

Корпоративные системы баз данных

© М.Л. Цымблер

22

Пример: создание кластера

```
CREATE CLUSTER p_sp
  (pid NUMBER);

CREATE TABLE p (
  pid NUMBER PRIMARY KEY, ...)
  CLUSTER p_sp (pid);

CREATE TABLE sp (
  pid NUMBER FOREIGN KEY REFERENCES p (pid), ...)
  CLUSTER p_sp (pid);
```

Корпоративные системы баз данных

© М.Л. Цымблер

23

Словарь данных

- *Словарь данных (database dictionary)* – общедоступный набор таблиц и представлений, в которых хранится различная системная информация (количество и имена таблиц и представлений, структура таблиц и др.). Словарь данных доступен только по чтению.
- Словарь данных содержит три набора таблиц, имеющих одинаковую структуру и различающихся префиксом имени:
 - **user_** объекты базы данных, созданные пользователем
 - **all_** все объекты, к которым имеет доступ пользователь
 - **dba_** все объекты, к которым имеют доступ все пользователи

Корпоративные системы баз данных

© М.Л. Цымблер

24

Пример: запросы к словарю данных

```
SELECT object_name, object_type  
FROM user_objects;
```

```
SELECT owner, object_name, object_type  
FROM all_objects;
```

```
SELECT table_name, column_name, data_type, data_length,  
       nullable, data_default  
FROM user_tab_columns  
ORDER BY table_name;
```

```
SELECT *  
FROM dba_users;
```

Корпоративные системы баз данных

© М.Л. Цымблер

25
