

**ORACLE®**



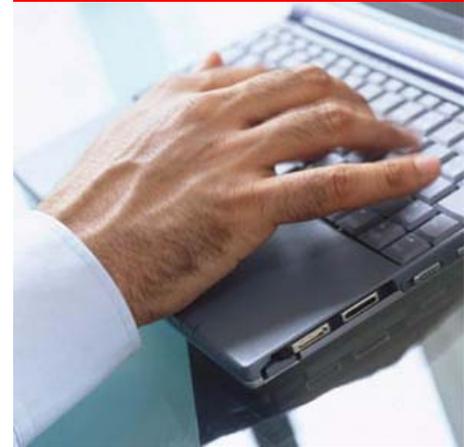
**ORACLE®**

# Oracle Database Change Notification

Игорь Мельников  
старший консультант

# План

- Слежение за изменениями в БД
- Обработка событий изменения данных в БД
- Поддержка в Oracle Provider for ADO DB.NET
- Новые возможности в Oracle Database 11g
- Демонстрация



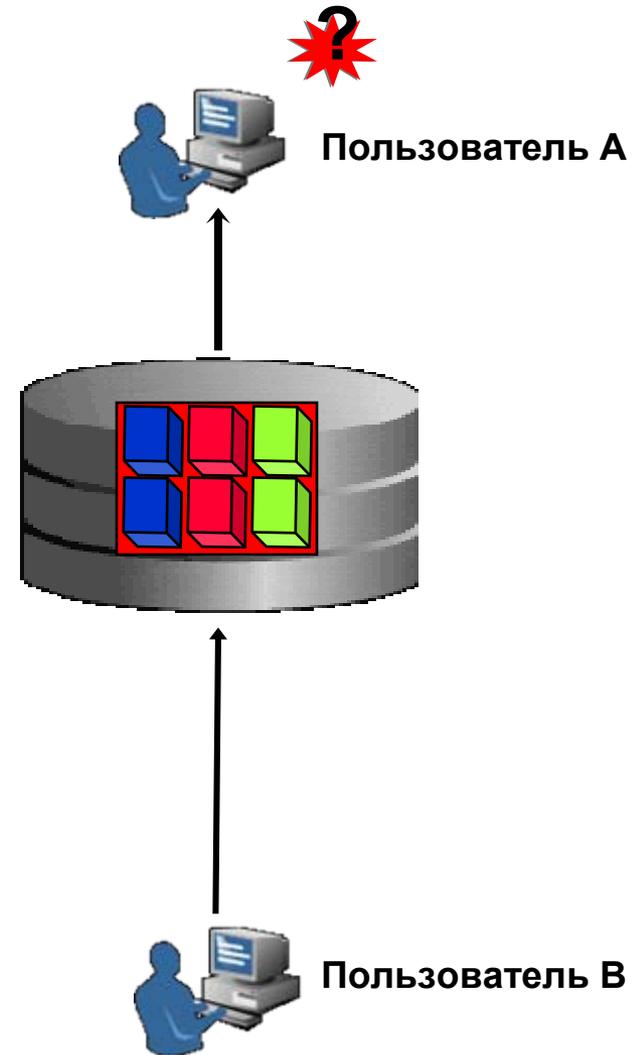


# Слежение за изменениями в БД

# Пример

## Частая ситуация в приложении

- Пользователь А: подключается к БД и запрашивает данные из таблицы X
- Пользователь А: эти данных поступают в приложение (в массив, глобальные переменные, в визуальные элементы управления (напр: Table Grid )
- Пользователь В: подключается к БД и модифицирует эти данные в таблице X
- **Пользователь А использует некорректные данные !**



# Решения

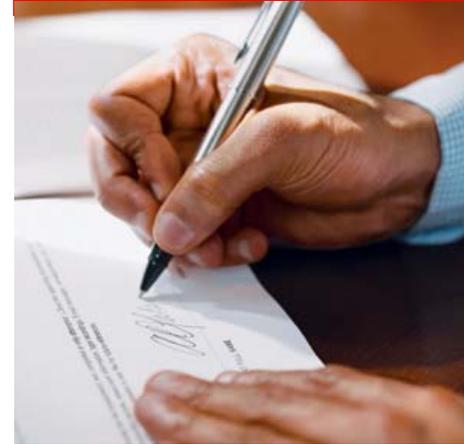
## *Проблемы несогласованности данных*

- Не кэшировать данные – всегда брать их из БД
  - Недостаток: Затраты ресурсов на постоянное обращение к СУБД
- Ввести дополнительную колонку в таблицу (время изменения/флаг изменения)
  - Недостаток: дополнительно место на хранение и обработку этого поля
- Клиент перед каждой операцией опрашивает СУБД на предмет прошедших изменений данных
  - Недостаток:
    - Значительное увеличение сетевого трафика
    - Потеря информации об изменениях которые произошли между опросами

# Идеальное решение

## *Слежение за изменениями*

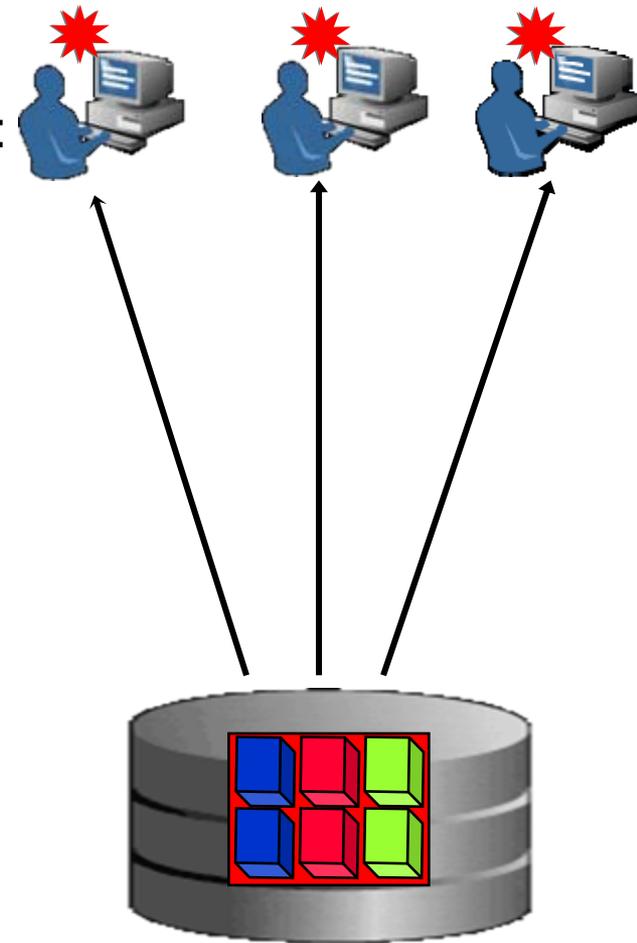
- Только СУБД должна уведомлять об изменениях
  - СУБД - единственный источник “истины”
- Как только произошло изменение СУБД должна сама уведомить всех заинтересованных клиентов
  - Обновлять кэш на клиенте когда это действительно необходимо
  - Клиенты могут использовать всегда актуальный кэш (особенно серверы приложений!)
- Сетевой трафик:
  - Трафик генерируется только в момент когда происходит изменение
  - Посылается не сами измененные данные, а только информация об изменении



# Введение в Oracle Database Change Notification

# Oracle Database Change Notification

- Когда происходит изменение данных СУБД отправляет клиентам уведомление
- Слежение за различными типами операций:
  - DML (insert/update/delete)
  - DDL (alter table add ..)
  - Старт/Останов (startup/shutdown)
  - Определение объекта слежение по зависимостям (Напр: через select)
- Для DML-операций - нотификация только после фиксации транзакции !
- Минимальные затраты на поддержку: работает на основе анализа оперативных журналов (не требуется режим *archive*log)



# Oracle Database Change Notification

## *Процесс слежения за изменениями*

- Пользователь должен иметь привилегию CHANGE NOTIFICATION
- Клиент “подписывается” на изменения – определяет объекты и регистрирует процедуру-обработчик (callback function)
- На клиенте запускается процесс слежения (change notification listener)
- Когда происходит изменение данных СУБД отправляет клиентам уведомление
- На клиенте автоматически выполняется процедура-обработчик

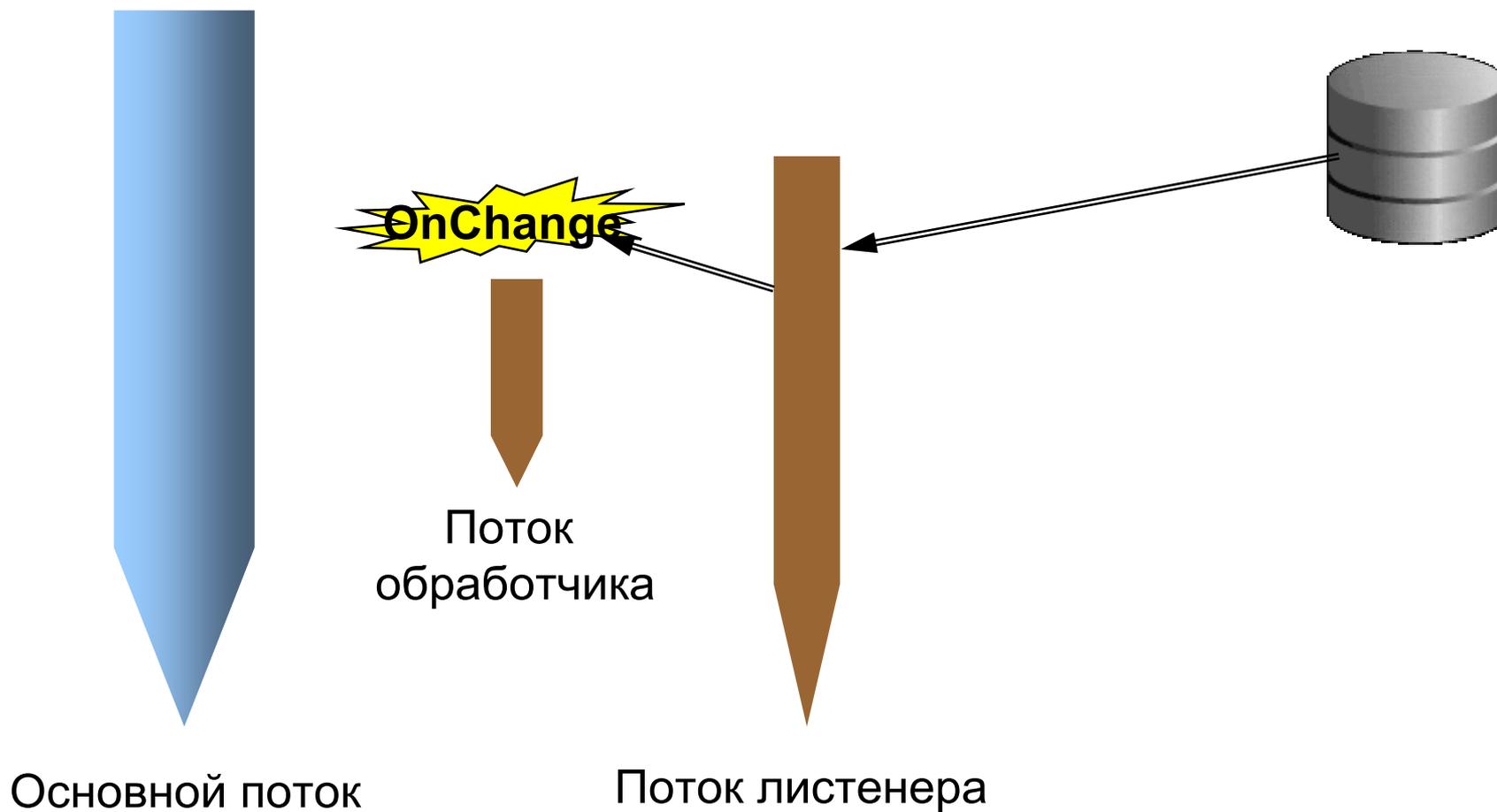
# Информация передаваемая клиенту

- Имя и схема объекта который был изменен
- Тип события, которое изменило данные
  - INSERT, UPDATE, DELETE, ALTER TABLE, or DROP TABLE
  - Глобальное событие, такое как STARTUP или SHUTDOWN
    - В RAC - когда первый экземпляр стартовал или последний был остановлен
- Для DML-операций: RowId строк, которые были модифицированы (не сами данные!)

# Типы запросов для слежения

- Все запросы:
  - К представлениям (views)
  - Соединения (join)
  - Запросы к табличным функциям
  - REF Cursors
- ...Исключая запросы использующие:
  - Системные таблицы и представления (напр: X\$, V\$)
  - Запросы включающие обращение к db-link
  - Материализованные представления
- Нотификация, если изменился любой объект от которого зависит запрос:

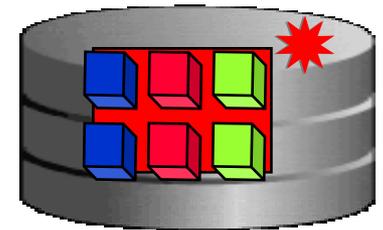
# Приложение стало многопоточным



# Server change notification

## *Обработчик изменений - на стороне СУБД*

- Обработчик изменений также возможен на сервере БД
- Определяемая разработчиком PL/SQL-функция, выполняемая на сервере
- Пакет DBMS\_CHANGE\_NOTIFICATION
  - Регистрация серверного обработчика
  - Регистрация PL/SQL-функции с определенной сигнатурой
- PL/SQL-callback срабатывает на сервере (1 раз на событие) асинхронно к сессии вызвавшей изменение



# Пакет DBMS\_CHANGE\_NOTIFICATION

## *Создание обработчика*

```
CREATE PROCEDURE my_callback(ntfnds in sys.chnf$_desc) is
  tbname          VARCHAR2(60);
  event_type      NUMBER;
  numtables       NUMBER;
  operation_type  NUMBER;
BEGIN
  numtables := ntfnds.numtables;
  event_type := ntfnds.event_type;

  IF (event_type = DBMS_CHANGE_NOTIFICATION.EVENT_OBJCHANGE) THEN
    FOR i IN 1..numtables LOOP
      tbname          := ntfnds.table_desc_array(i).table_name;
      operation_type := ntfnds.table_desc_array(I).Opflags;
      ...
    END LOOP;
  END IF;
END;
```

# Пакет DBMS\_CHANGE\_NOTIFICATION

## *Регистрация обработчика*

```
DECLARE
    regds      sys.chnf$_reg_info;
    regid      number;
    v_xId      number;
    qosflags   number;
BEGIN
    qosflags := DBMS_CHANGE_NOTIFICATION.QOS_RELIABLE +
    DBMS_CHANGE_NOTIFICATION.QOS_ROWIDS;

    REGDS := SYS.CHNF$_REG_INFO ('my_callback', qosflags, 0,0,0);
    regid := DBMS_CHANGE_NOTIFICATION.NEW_REG_START (REGDS);

    SELECT id INTO v_xId FROM companies WHERE id = 27364;

    DBMS_CHANGE_NOTIFICATION.REG_END;
END;
```



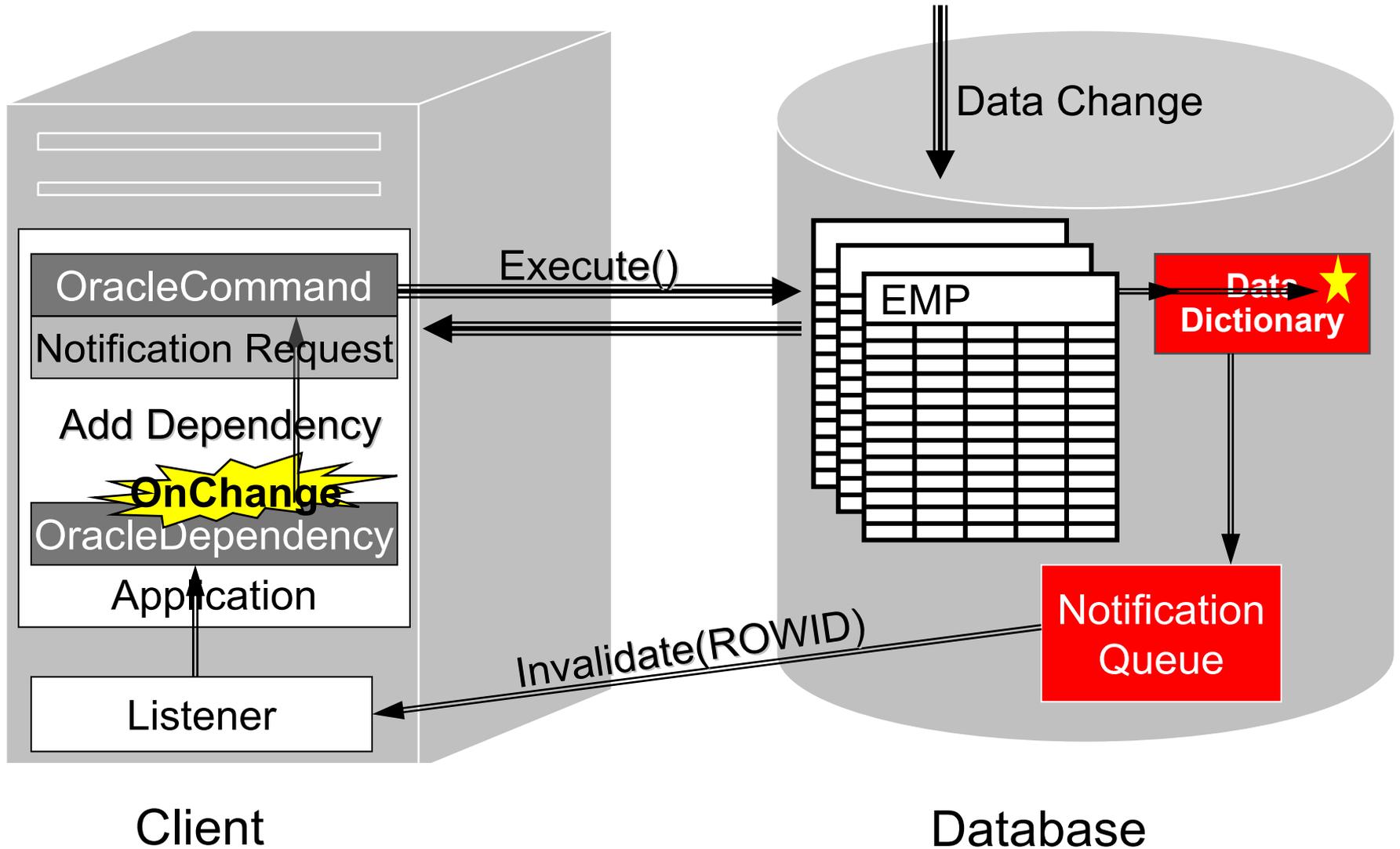
# Поддержка в Oracle Data Provider for ADO DB.NET

# Oracle Data Provider for ADO DB.NET

## *Требования*

- Версии
  - Oracle Database 10.2 и выше
  - Oracle Client 10.2 и выше
  - ODP.NET 10.2 и выше
- На клиенте должен быть открыт порт
  - ODP.NET выбирает порт случайно
  - Либо номер порта можно задать явно (программно)
  - Запускает один прослушиватель (listener) на домен приложений
  - Прослушиватель запускается только после первой регистрации
  - Прослушиватель - просто отдельный поток в приложении (не сервис !) – запускается автоматически в момент регистрации первой change notification callback-функции

# Change Notification in ODP.NET



# Пример регистрации в ODP.NET

*Object Pascal (компилятор RemObjects Chrome)*

```
var
  v_xConn      : OracleConnection
  v_xCommand   : OracleCommand;
  v_xDependency : OracleDependency;
begin
  ... ..
  v_xCommand   := new OracleCommand('select * from demo',
                                     v_xConn);
  v_xDependency := new OracleDependency(v_xCommand);

  v_xCommand.Notification.IsNotifiedOnce := false;
  v_xCommand.AddRowid := true;
  v_xDependency.OnChange += OnDatababaseNotification;

  v_xCommand.ExecuteNonQuery;
end;
```

# Пример обработчика

```
method MyClass.OnDatababaseNotification(  
                                Src   : TObject;  
                                Args  : OracleNotificationEventArgs);  
Begin  
    if Args.Source = OracleNotificationSource.Data then  
        Begin  
            if Args.Info = OracleNotificationInfo.Insert Then  
                Begin  
                    Var v_xRowId OracleParameter = new OracleParameter();  
  
                    //получаем rowid НОВОЙ строки  
                    v_xRowId.Value := args.Details.Rows[0]['rowid'].toString;  
  
                    ... ..  
                End;  
            End;  
        End;  
    End;  
End;
```

# Change Notification в ODP.NET

## *Тонкая настройка*

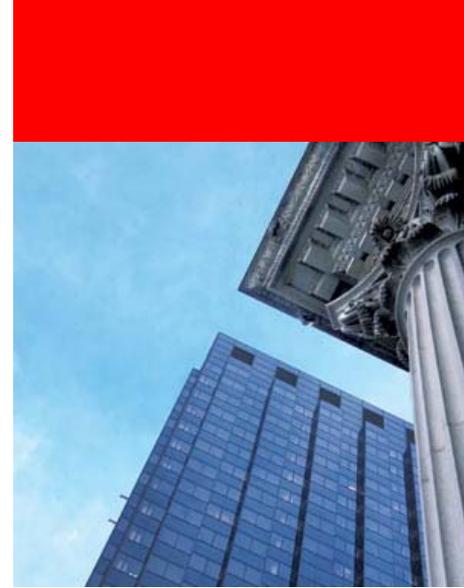
- “Отписаться” от уведомлений
  - `OracleDependency.RemoveRegistration`
- Сколько времени следить за изменениями
  - `OracleCommand.Notification.Timeout`
- Подписаться на одно уведомление или получать их постоянно:
  - `OracleCommand.Notification.IsNotifiedOnce = false`
- Группировать объекты слежения в один обработчик
  - `OracleDependency.AddCommandDependency`
  - Все сообщения об изменениях будут “запускать” один обработчик

# Change Notification в ODP.NET

## *Тонкая настройка (продолжение)*

- Сохранять сообщения в памяти сервера (по умолчанию) или сохранять в БД (на диске)
  - Защита от потери сообщений – для гарантированной доставки
  - `OracleCommand.Notification.IsPersistent`

# НОВЫЕ ВОЗМОЖНОСТИ В Oracle Database 11g



ORACLE®  
DATABASE 11<sup>g</sup>

ORACLE®

# Database Change Notification

## *Поддержка Java-разработчиков*

- Полная поддержка Database Change Notification в JDBC-драйвере версии 11.1

# Query Result Change Notification

## *Слежение за изменением выборки*

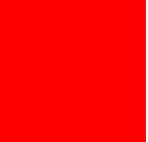
- Слежение за изменением результата выборки (*select*), а не объектов, от которых она зависит (как в 10gR2)
- Обычный и гарантированный режим
  - Сообщение, только если выборка изменилась
  - Например (не посылать сообщение):

```
UPDATE emp SET salary = salary + 10 WHERE empno = 111;  
UPDATE emp SET salary = salary - 10 WHERE empno = 111;  
COMMIT;
```
  - На оператор *select* накладываются ограничения
    - Нет агрегатных функций
    - Поля только NUMBER и VARCHAR2 ... ..
- Управление пакетом и частотой посылки сообщений
  - Например: посылать уведомление после 5 транзакций

# Query Result Change Notification

## *Продолжение*

- Новый пакет DBMS\_CQ\_NOTIFICATION
- Пока обработка только на стороне сервера (PL/SQL-callback)
- Пока ODP.NET и JDBC не поддерживают QRCN



**ORACLE®**

**Демонстрация**

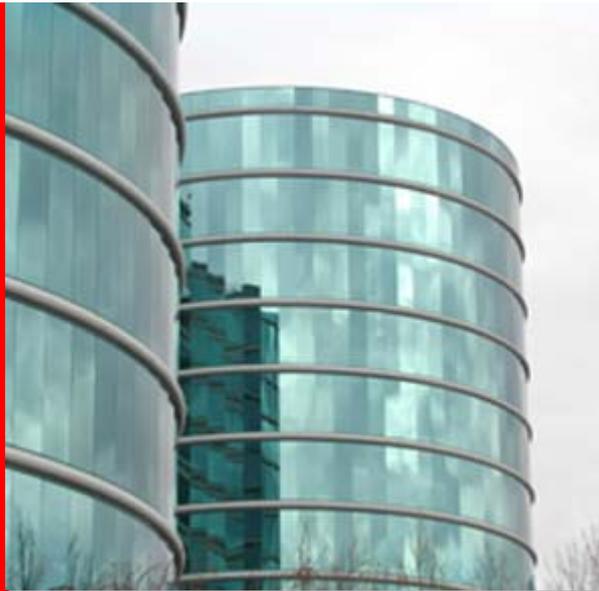
# **Oracle Database Change Notification**

# Oracle Database Change Notification

## *Заключение*

- Готовое и полнофункциональное решение для замены в приложениях логики проверки актуальности данных
- Применяется для контроля над данными которые больше читаются, и относительно редко меняются
- Для приложений имеющих свои механизмы кэширования
- Очень актуально для серверов приложений !

Q&A



**ORACLE®**

Игорь Мельников  
Старший консультант Oracle СНГ

Email : [Igor.Melnikov@oracle.com](mailto:Igor.Melnikov@oracle.com)  
Phone : +7 (495) 641 14 00  
Direct: +7 (495) 641 14 42  
Mobile: +7 (915) 205 26 27