

|  |
| --- |
| EXTRACT, TRANSFORM, LOAD  Labwork 10  Basic Parallel Execution |

Оглавление

[1 ParaLlel select 3](#_Toc500276475)

[2 ParalLel DDL 5](#_Toc500276476)

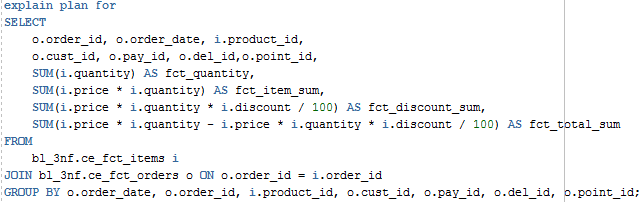
[3 Parallel DML 6](#_Toc500276477)

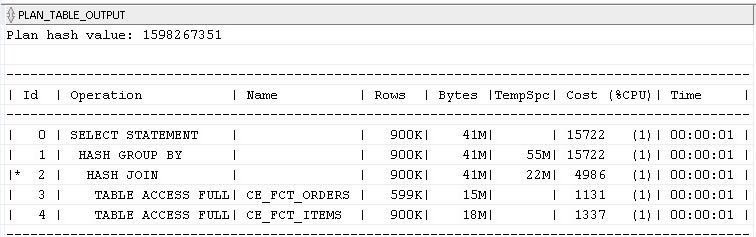
# ParaLlel select

Все скрипты находятся на Git в папке [BI-Lab-2017](https://github.com/mikitakandratsiuk/BI-Lab-2017)/[\_2. ETL](https://github.com/mikitakandratsiuk/BI-Lab-2017/tree/master/_2.%20ETL)/[Tasks](https://github.com/mikitakandratsiuk/BI-Lab-2017/tree/master/_2.%20ETL/Tasks)/[Maryna\_Hlazunova](https://github.com/mikitakandratsiuk/BI-Lab-2017/tree/master/_2.%20ETL/Tasks/Maryna_Hlazunova)/Task 10/. Эта директория является домашней для данной лабораторной работы и далее упоминаться не будет.

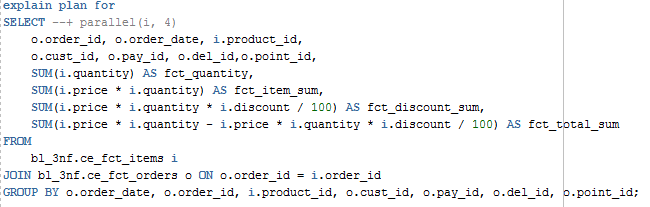
Для увеличения скорости выполнения запросов зачастую используют хинт параллельного выполнения parallel.

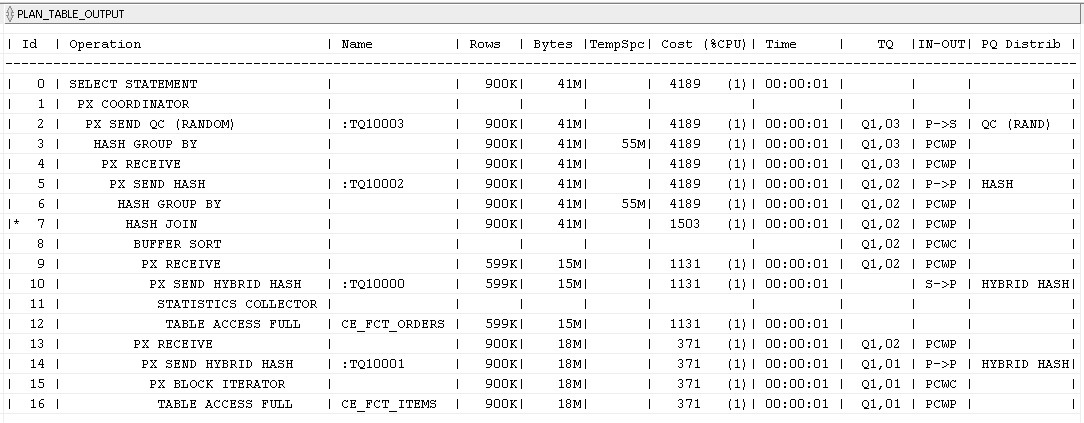
Проанализируем выборку одних и тех же данных с этим хинтом и без него.





А теперь подставим хинт parallel и посмотрим план выполнения запроса в данном случае.

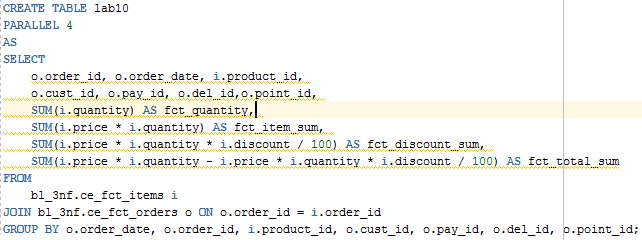




Можно сделать вывод, что при использовании распараллеливания уменьшается cost выполнения запроса.

# ParalLel DDL

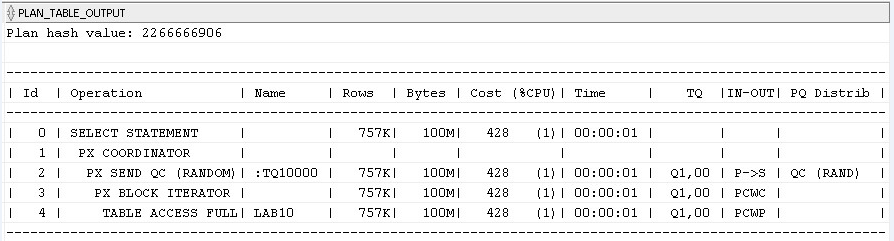
Параллельность можно также использовать в операциях DDL, например, явно указывать при создании таблицы основанный на методе CTAS:



В таком случае при обычном обращении к этой таблице будет автоматически выполняться параллельное выполнение:



План запроса:

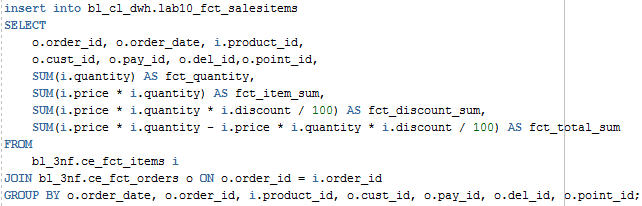


# Parallel DML

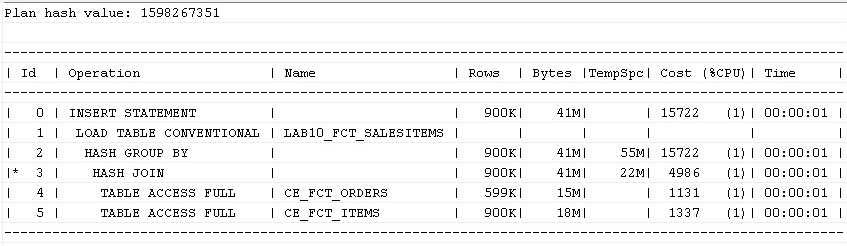
Также очень выгодно распараллеливание при операциях DML, например, при вставке записей или их обновлении.

Рассмотрим вариант вставки данных в нашу фактовую таблицу. Таким образом мы и часть таска сделаем и проверим эффективность хинта parallel на построенном DWH.

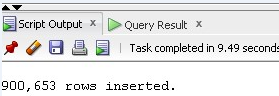
Вначале рассмотрим вставки данных без учета распараллеливания:



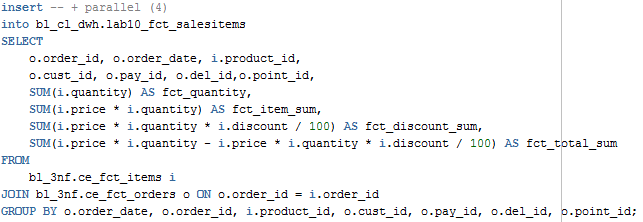
План выглядит следующим образом:



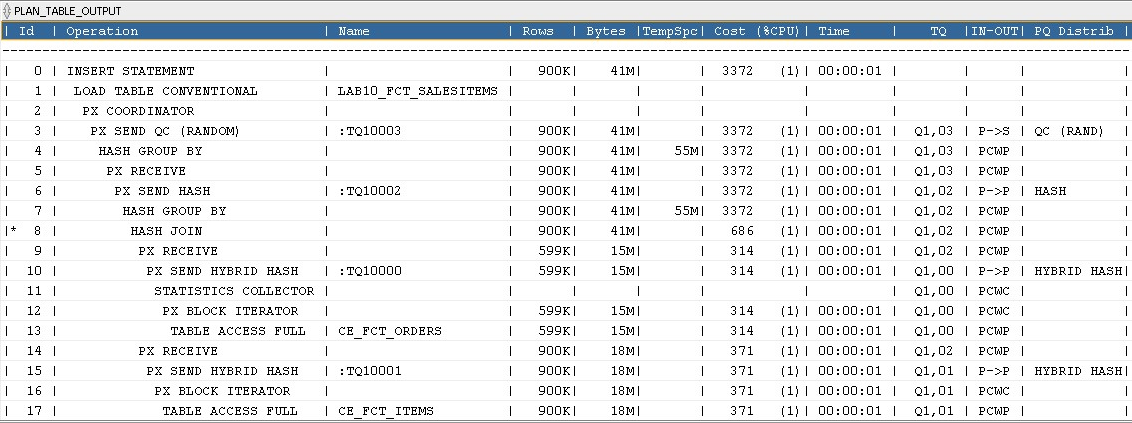
Вставка данных происходит за 9,49 секунд.



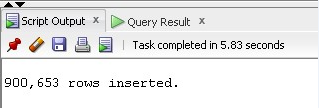
Далее делаем такие же действия, но уже с использованием parallel хинта:



План выполнения:



Время вставки такого же количества записей было выполнено за 5,83 секунды.



Таким образом, применение хинта parallel привело к уменьшению времени вставки в 1,8 раза, и уменьшению cost почти в 4 раза. Из этого можно сделать вывод, что использование распараллеливания выгодно в нашем случае.