team17_assignment3_report1

Member：106070031 宋友友, 107020024 謝柏陞, 107020005 黃煒翔

實作

Parser

- Parser 裡面增加變數 `public static boolean IS_EXPLAIN;`，並在 new Parser 的時候對其初始化為 false
- 利用 lexer 的 matchKeyword("explain") ，在 Parser.queryCommand() 跟 Parser.updateCommand() 裡面加上 EXPLAIN 的判別
- 並在 Parser 新增一個 isExplain()，這樣 Planner 就能呼叫 Parser.isExplain() 來獲得是否執行 EXPLAIN 的資訊
- 例如本來 Planner 要呼叫 `qPlanner.createPlan(data, tx);` ，如果要 Explain 的話，就變成呼叫 `qPlanner.ExplainCreatePlan(data, tx);`

這樣做的優缺點：

優點

1. Verifier 不用改
2. QueryData 跟 XXXXData 不用改

因為資料型態跟先前一樣

缺點

1. Planner 要針對 EXPLAIN 增加對應的 function，不能沿用本來的 function
2. QueryPlanner 跟 UpdatePlanner 這兩個 interface 都要改，影響了後面的 As3QueryPlanner 跟 As3UpdatePlanner 的實作，且沒辦法透過修改 vanilladb.properties 來切換 QueryPlanner 跟 UpdatePlanner

Planner

- Planner.java:透過parser裡的isExplain()來確認現在是否需要EXPLAIN，若為要EXPLAIN，對於plaaner裡的function就呼叫explain的版本。
- QueryPlanner:在最後呼叫ExplainPlanner，也就是在整個plan tree得最上面加一層ExplainPlan。之後return ExplainPlan。

- UpdatePlanner:因為UpdatePlanner裡只有Insert、Delete、Modify會用到plan，所以只有在這幾個function裡呼叫ExplainPlan。

ExplainPlan

- 把explainPlan包在原有的algebra tree的最外面，如此一來可以存取下面所有subPlan的各項數據，而我們implement `EXPLAIN` 的作法:
  1. 使用遞迴的使用 `toString` 來生成一個output string，再藉由 `VarCharConstant` 把string轉為constant obj，因此下面每一個plan都新增了該層的 `toString`，會各自回傳該層的資訊。
  2. 在schema中增加一個query-plan
  3. Open中會創建一個UpdateScan的TempTable，將底下plan的open()回傳的scan複製下來，再把output的constant加到scan的query-plan裡面。

測試

1. A query accessing single table with WHERE

- `EXPLAIN SELECT c_d_id FROM customer WHERE c_id<2`

```
c_d_idquery-plan
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------     1
  ->ProjectPlan (#blks=30001, #recs=0)
    ->SelectPlan  pred:(c_id<2.0) (#blks=30001, #recs=0)
      ->TablePlan on (customer) (#blks=30001, #recs=60000)
Actual #recs: 0
```

1. A query accessing multiple tables with WHERE

- `EXPLAIN SELECT d_id FROM district, warehouse WHERE d_w_id = w_id;`

```
d_idquery-plan
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------     1
  ->ProjectPlan (#blks=43, #recs=40)
    ->SelectPlan  pred:(d_w_id=w_id)  (#blks=43, #recs=40)
```

```
      ->ProductPlan (#blks=43, #recs=40)
        ->TablePlan on (district) (#blks=3, #recs=20)
        ->TablePlan on (warehouse)  (#blks=2, #recs=2)
Actual #recs: 40
```

1. A query with ORDER BY

- `EXPLAIN SELECT c_d_id FROM customer WHERE c_id<2 ORDER BY c_id`

```

```

1. A query with GROUP BY and at least one aggregation function (MIN, MAX, COUNT, AVG... etc.)

- `EXPLAIN SELECT COUNT(d_id) FROM district, warehouse WHERE d_w_id = w_id GROUP BY w_id`

```
countofd_idquery-plan
------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
------------------------------------------------------------------------    40
  ->ProjectPlan (#blks=7, #recs=1)
    ->GroupByPlan: (#blks=7, #recs=1)
      ->SortPlan (#blks=7, #recs=40)
        ->SelectPlan  pred:(d_w_id=w_id)  (#blks=43, #recs=40)
          ->ProductPlan (#blks=43, #recs=40)
            ->TablePlan on (district) (#blks=3, #recs=20)
            ->TablePlan on (warehouse)  (#blks=2, #recs=2)
Actual #recs: 1
```

小結

有一點 bug：

- 開頭不是 query-plan，是 countofd_id