Database Systems Assignment 3 Report1 - Team20

> Members: 資工大三 108062208 阮棠

Implementation

Lexer / Parser / QueryData

> Commit: e106122e - Feat: Add keyword in lexer, parse, mod qdata

- Lexer: 在keyword list裡面新增`explain`
- Parser: queryCommand() 最前面看是不是有match到`explain`，有則新建explainFields(雖然只有一個Field，參考其他command的作法)，增加`query-plan` field，傳進QueryData的constructor
- QueryData: 增加`explainFields`

toString

> Commit: c0945254 - Feat: impl toString for plans

- Plan interface: 新增`toString` method
- 各個Plan: 實作`toString`，每個Plan都會呼叫child的`toString`，直到TablePlan(no more children)。每次Plan.toString會把小孩的String切開來，再增加新的tab，則可遞迴層層印出所有經過的Plan和想要的內容

Planners

> Commit: dbb11935 - Feat: Add Explain Plan proc when creating Qplan

- import ExplainPlan in BasicQueryPlanner
- 在CreatePlan的最後面加上Step 7: Create ExplainPlan if requested
- 參考SortPlan的寫法，如果傳過來的QueryData裡面explainfields不為空（有`query-plan`），則進入ExplainPlan
    - 回傳的Plan也是ExplainPlan

ExplainPlan / ExplainScan

> Commit: 17a01aae - Feat: Impl explain plan & scan that works (v0)

ExplainPlan

- 建立Schema: 只有一個field: query-plan ，type 是 `VARCHAR(500)`

```
public ExplainPlan(Plan p, Set<String> fldNames, \
Transaction tx) {
    this.p = p;
    this.tx = tx;
    for (String fldname : fldNames)
        schema.addField(fldname, VARCHAR(500));
    }
```

Temptable

- 用Materialize的Temptable實作explain
- 原本打算建立新的table再drop，但也不知道drop的時機，應該還是存temp比較合理
- 參考 Update 的 UpdateScan 和 TablePlan / TableScan 的寫法
- 根據Schema建立TempTable之後先insert一個新record，再把值（`toString`跑出來的東西）塞進那個record

```
ctor {
        ...

    expTempTable = new TempTable(schema, this.tx);
    expTableScan = (TableScan) expTempTable.open();
    expTableScan.insert();
    expTableScan.setVal("query-plan", \
        (Constant) new VarcharConstant(toString()));
    }
```

Actual records

> Commit: dee2c94a - Feat: Count actual records with result set (scan)

- In `recordsOutput`
- 需要回傳真正 SELECT 出來的東西有幾個records
- 把傳入ExplainPlan的Plan(最後一個Plan, e.g. ProjectPlan)打開得到Scan
- 呼叫scan.next() 再數有幾個就是record數量
- 會被`toString`呼叫，這樣就可以印出Actual records

ExplainScan

- 參考ProjectScan的寫法
- 把傳進來的Scan再包裝一次
- 讓resultSet可以抓到做出來的資料

Output format adjustment

> Commit: b2c1b5f7 - Fix: Adjust print order to look like spec

1. Explain時第一列會消失
    - SQL console 遇到 explain 的時候不會在----後面印一個換行，所以字會跑上去
    - 處理：在ExplainPlan 的 `toString` 最前面加換行
2. TablePlan 的順序和 SPEC 相反
    - 在 ProductPlan 的 `toString` 先印 cs2 再印 cs1 就會和 SPEC 上一樣了

Code trace

How SQL Console proceeds `EXPLAIN`

```
Top down
SQL console EXPLAIN
-> doquery
-> rs = executeQuery
-> RemoteStatementImpl
-> pln = VanillaDb.newPlanner().createQueryPlan(qry.tx)
-> Planner.createQueryPlan (return Plan)
-> new RemoteResultSetImpl(pln, rconn);
-> return a pointer to a RemoteResultSetImpl
```

Tracing and experimenting for implementation part

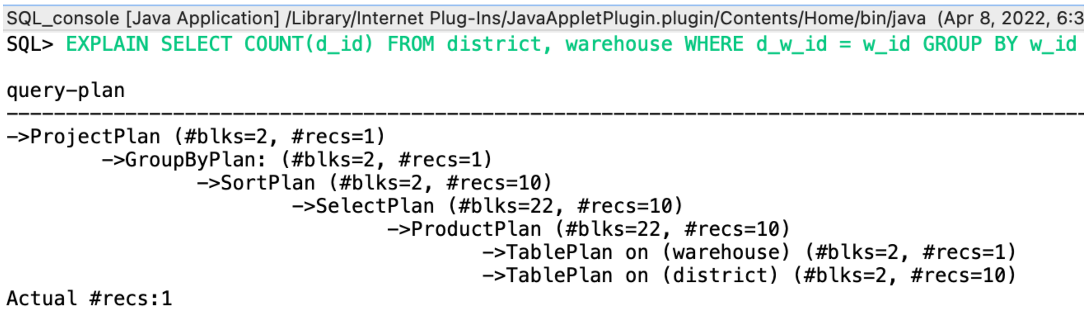
- toString
    - Trace的時候在找怎麼把所有child都印出來，還在用 `System.out.println` 實驗
    - 看到 MaterializePlan 裡面有寫好的`toString`, 層層呼叫child(都是Plan)的`toString`，就可以把全部都印出來
    - 把 `toString` 加進 Plan interface
- TempTable
    - Used in SortPlan

Experiments

> Testing SQL commands inspired by ref[4] and TpccSchemaBuilderProcParamHelper

Sample of As3 SPEC

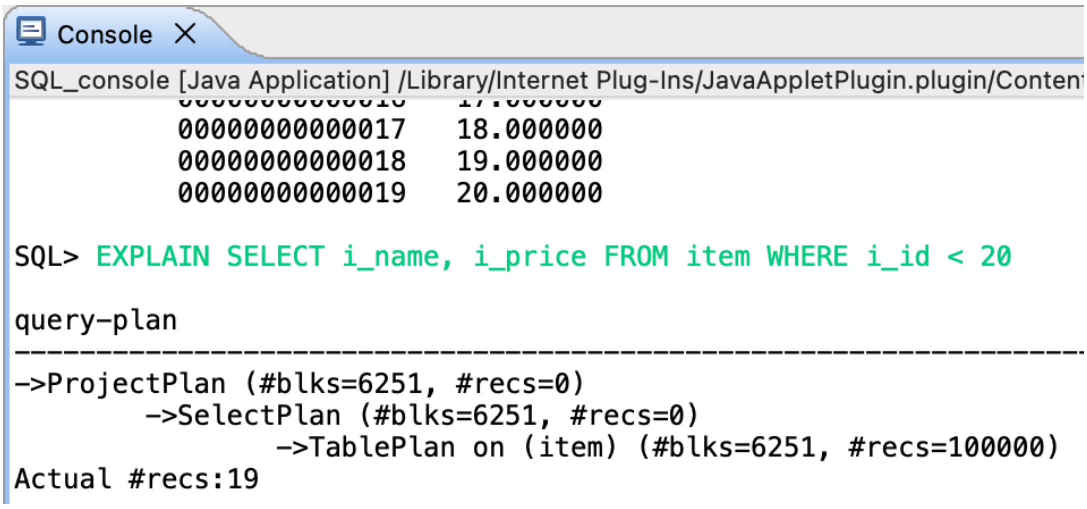
```
EXPLAIN SELECT COUNT(d_id) FROM district, warehouse \
WHERE d_w_id = w_id GROUP BY w_id
```



```
SQL_console [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 6:3
SQL> EXPLAIN SELECT COUNT(d_id) FROM district, warehouse WHERE d_w_id = w_id GROUP BY w_id

query-plan
----------------------------------------------------------------------------------------------
->ProjectPlan (#blks=2, #recs=1)
        ->GroupByPlan: (#blks=2, #recs=1)
                ->SortPlan (#blks=2, #recs=10)
                        ->SelectPlan (#blks=22, #recs=10)
                                ->ProductPlan (#blks=22, #recs=10)
                                        ->TablePlan on (warehouse) (#blks=2, #recs=1)
                                        ->TablePlan on (district) (#blks=2, #recs=10)
Actual #recs:1
```

A query accessing single table with WHERE

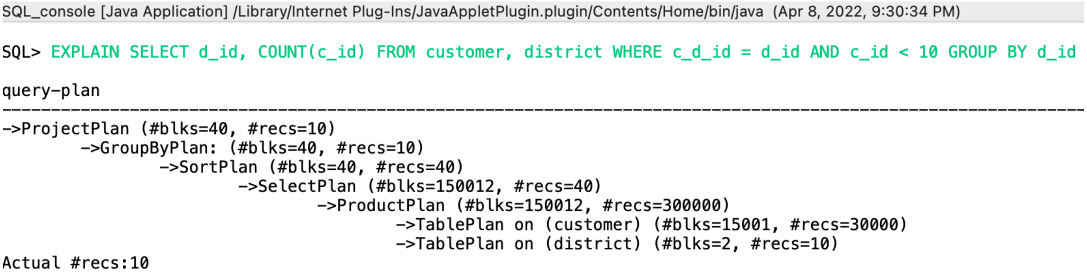
```
EXPLAIN SELECT i_name, i_price FROM item WHERE i_id < 20
```



```
SQL_console [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Content
             00000000000016    17.000000
             00000000000017    18.000000
             00000000000018    19.000000
             00000000000019    20.000000

SQL> EXPLAIN SELECT i_name, i_price FROM item WHERE i_id < 20

query-plan
-------------------------------------------------------------------
->ProjectPlan (#blks=6251, #recs=0)
        ->SelectPlan (#blks=6251, #recs=0)
                ->TablePlan on (item) (#blks=6251, #recs=100000)
Actual #recs:19
```

A query accessing multiple tables with WHERE

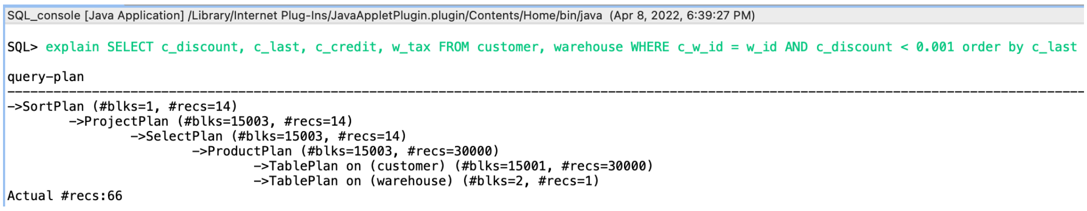
```
EXPLAIN SELECT d_id, COUNT(c_id) FROM customer, district \
WHERE c_d_id = d_id AND c_id < 10 GROUP BY d_id
```



```
SQL_console [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 9:30:34 PM)

SQL> EXPLAIN SELECT d_id, COUNT(c_id) FROM customer, district WHERE c_d_id = d_id AND c_id < 10 GROUP BY d_id

query-plan
--------------------------------------------------------------------------------------------------------------
->ProjectPlan (#blks=40, #recs=10)
        ->GroupByPlan: (#blks=40, #recs=10)
                ->SortPlan (#blks=40, #recs=40)
                        ->SelectPlan (#blks=150012, #recs=40)
                                ->ProductPlan (#blks=150012, #recs=300000)
                                        ->TablePlan on (customer) (#blks=15001, #recs=30000)
                                        ->TablePlan on (district) (#blks=2, #recs=10)
Actual #recs:10
```

A query with ORDER BY

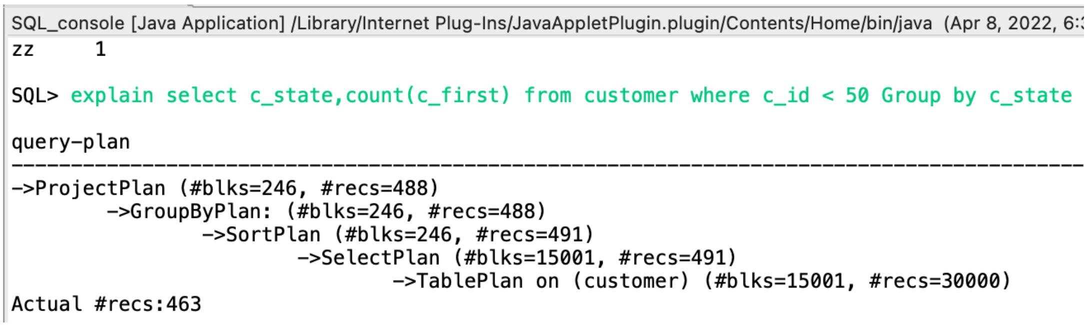
```
EXPLAIN SELECT c_discount, c_last, c_credit, w_tax FROM customer, warehouse \
WHERE c_w_id = w_id AND c_discount < 0.001 ORDER BY c_last
```



```
SQL_console [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 6:39:27 PM)

SQL> explain SELECT c_discount, c_last, c_credit, w_tax FROM customer, warehouse WHERE c_w_id = w_id AND c_discount < 0.001 order by c_last

query-plan
----------------------------------------------------------------------------------------------------------------------------------------
->SortPlan (#blks=1, #recs=14)
        ->ProjectPlan (#blks=15003, #recs=14)
                ->SelectPlan (#blks=15003, #recs=14)
                        ->ProductPlan (#blks=15003, #recs=30000)
                                ->TablePlan on (customer) (#blks=15001, #recs=30000)
                                ->TablePlan on (warehouse) (#blks=2, #recs=1)
Actual #recs:66
```

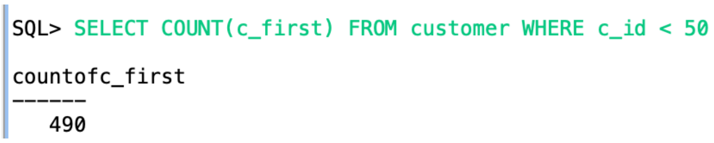
A query with GROUP BY and at least one aggregation function (MIN, MAX, COUNT, AVG... etc.)

```
EXPLAIN SELECT c_state,COUNT(c_first) FROM customer WHERE c_id < 50 \
GROUP BY c_state
```

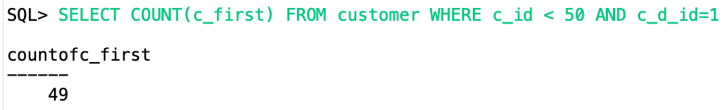


```
SQL_console [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 6:3
zz      1

SQL> explain select c_state,count(c_first) from customer where c_id < 50 Group by c_state

query-plan
------------------------------------------------------------------------------------------
->ProjectPlan (#blks=246, #recs=488)
        ->GroupByPlan: (#blks=246, #recs=488)
                ->SortPlan (#blks=246, #recs=491)
                        ->SelectPlan (#blks=15001, #recs=491)
                                ->TablePlan on (customer) (#blks=15001, #recs=30000)
Actual #recs:463
```

- 這裡可以發現c_id不是primary key，使用 WHERE c_id < 50 會有超過49個，會選出每個district的49個再GROUP BY c_state



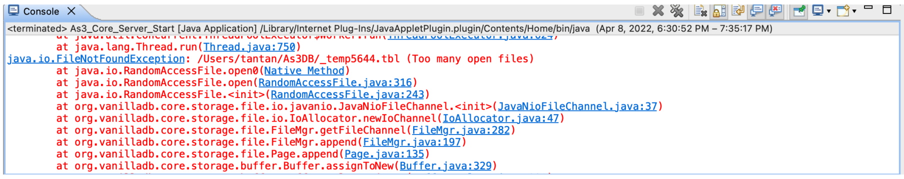
```
SQL> SELECT COUNT(c_first) FROM customer WHERE c_id < 50

countofc_first
------
   490
```



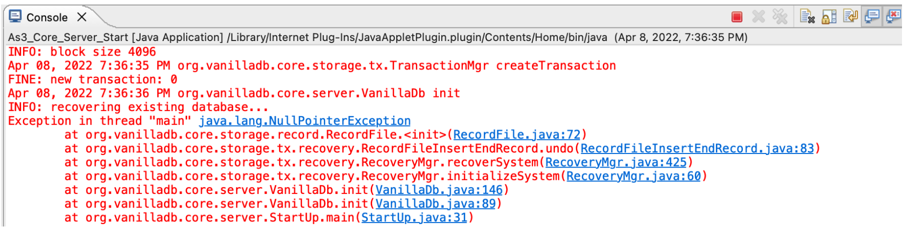
```
SQL> SELECT COUNT(c_first) FROM customer WHERE c_id < 50 AND c_d_id=1

countofc_first
------
     49
```

Too many open files

```
SELECT AVG(c_discount) FROM customer \
WHERE c_discount < 0.1 GROUP BY c_credit

SELECT d_id, COUNT(c_id) FROM customer, district \
WHERE c_d_id = d_id AND c_id < 1000 GROUP BY d_id
```



```
<terminated> As3_Core_Server_Start [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 6:30:52 PM – 7:35:17 PM)
        at java.lang.Thread.run(Thread.java:750)
java.io.FileNotFoundException: /Users/tantan/As3DB/_temp5644.tbl (Too many open files)
        at java.io.RandomAccessFile.open0(Native Method)
        at java.io.RandomAccessFile.open(RandomAccessFile.java:316)
        at java.io.RandomAccessFile.<init>(RandomAccessFile.java:243)
        at org.vanilladb.core.storage.file.io.javanio.JavaNioFileChannel.<init>(JavaNioFileChannel.java:37)
        at org.vanilladb.core.storage.file.io.IoAllocator.newIoChannel(IoAllocator.java:47)
        at org.vanilladb.core.storage.file.FileMgr.getFileChannel(FileMgr.java:282)
        at org.vanilladb.core.storage.file.FileMgr.append(FileMgr.java:197)
        at org.vanilladb.core.storage.file.Page.append(Page.java:135)
        at org.vanilladb.core.storage.buffer.Buffer.assignToNew(Buffer.java:329)
```



```
As3_Core_Server_Start [Java Application] /Library/Internet Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java  (Apr 8, 2022, 7:36:35 PM)
INFO: block size 4096
Apr 08, 2022 7:36:35 PM org.vanilladb.core.storage.tx.TransactionMgr createTransaction
FINE: new transaction: 0
Apr 08, 2022 7:36:36 PM org.vanilladb.core.server.VanillaDb init
INFO: recovering existing database...
Exception in thread "main" java.lang.NullPointerException
        at org.vanilladb.core.storage.record.RecordFile.<init>(RecordFile.java:72)
        at org.vanilladb.core.storage.tx.recovery.RecordFileInsertEndRecord.undo(RecordFileInsertEndRecord.java:83)
        at org.vanilladb.core.storage.tx.recovery.RecoveryMgr.recoverSystem(RecoveryMgr.java:425)
        at org.vanilladb.core.storage.tx.recovery.RecoveryMgr.initializeSystem(RecoveryMgr.java:60)
        at org.vanilladb.core.server.VanillaDb.init(VanillaDb.java:146)
        at org.vanilladb.core.server.VanillaDb.init(VanillaDb.java:89)
        at org.vanilladb.core.server.StartUp.main(StartUp.java:31)
```

- 可能是因為TPC-C的特定資料太多，SortPlan沒辦法在我的電腦上打開這麼多TempTable需要的files
- Clone 了一份完全沒改過的也會有這個error

Feedback and future work

- 在其他電腦上測試 (e.g. Linux server)，想要測試在其他電腦會不會有太多open file的問題
- 應該另外開一個branch，才方便測試是不是有壞掉的功能(跟完全沒改的比較)
- 如果要再做其他SQL的explain的話應該就是要包一下updatecommand的東西，讓explain Plan可以讀到
- Trace Storage Engine，不然 `TempRecordFile` 都看不太懂

References

[1] Lecture slides

[2] Vanilla DB documentation

[3] FAQ

[4] https://www.tpc.org/tpc_documents_current_versions/pdf/tpc-c_v5.11.0.pdf

Link of this MD

https://hackmd.io/@tantan3141/Hkn7hjdm5