geth-query数据库优化

> 当geth-query重放合约后，需要将抽取出来的自定义合约运行中间数据导出到csv中，由于csv文件大小较大，因此对于导出数据的管理需要导入到postgres数据库中。但是由于数据量超过TB，因此需要对数据库中的表做查询优化。

总述

该部分一共有3步：

1. 对导出数据进行校验。一共有3种类型的校验。1. 在重放数据过程中校验from余额的正确性。2. 校验每个表中区块数目的正确性。3. 校验区块数目是否是连续的(check_data_integrity.py)。
2. 在postgres种建立每100W个块分表，然后将csv数据导入到postgres数据库中，第二次校验数据(后面两项)，数据校验通过后开始删除5张表中的间隔符(import_csv_to_pg.py)。
3. 在5张表之上建立索引(create_index.sql)。

优化

一共进行两种优化：

1. 建立B-tree索引，对于blockNumber直接建立索引，对于blockHash和txHash对于其前6个字符建立索引(包括0x)。
2. 对于索引建立聚类，将相同聚类的索引项对应的内存存放在一起。
3. 分表优化，对于transfers, events, traces3种表的数据量都比较大，可以尝试对其先分表，再导入数据，再检查数据完整性，然后建索引，最后建聚类。

优化结果

blocks

8500000 rows

区块	行数	大小
1-100W	100W	385M 428M
100W-200W	100W	385M 428M
200W-300W	100W	359M 402M
300W-400W	100W	359M 402M
400W-500W	100W	360M 403M
500W-600W	100W	364M 407M
600W-700W	100W	367M 410M
700W-800W	100W	380M 423M
800W-850W	100W	192M 214M

blockNumber

```
explain analyse select * from blocks where blockNumber=6810068;

 Planning Time: 0.203 ms
 Execution Time: 0.050 ms
```

blockHash

6810068 0x2e25423238ef2178c735747481fe40d2ac0002c13719c37369856fecf7cb56c2

```
explain analyse select * from blocks where left(blockHash,6)='0x2e25';

 Planning Time: 0.768 ms
 JIT:
   Functions: 20
   Options: Inlining false, Optimization false, Expressions true, Deforming true
   Timing: Generation 4.217 ms, Inlining 0.000 ms, Optimization 0.000 ms,
Emission 0.000 ms, Total 4.217 ms
 Execution Time: 4.677 ms
```

transactions

536061133 rows

区块	行数	大小
1-100W	1674262	1040M 1156M
100W-200W	6383128	3285M 3751M
200W-300W	7305457	4178M 4702M
300W-400W	20971627	13G 15G
400W-500W	113525623	80G 88G
500W-600W	124040734	97G 108G
600W-700W	95916489	77G 89G
700W-800W	108960721	91G 101G
800W-850W	57283092	49G 54G

blockNumber

```
explain analyse select * from transactions where blockNumber=6810068;

 Planning Time: 8.694 ms
 Execution Time: 5.106 ms
```

blockHash

6810068 0x2e25423238ef2178c735747481fe40d2ac0002c13719c37369856fecf7cb56c2

```
explain analyse select * from transactions where left(blockHash,6)='0x2e25';

 Planning Time: 1.447 ms
 JIT:
   Functions: 20
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 4.403 ms, Inlining 0.000 ms, Optimization 0.000 ms, Emission 0.000 ms, Total 4.403 ms
 Execution Time: 29.343 ms
```

transactionHash

```
explain analyse select * from transactions where left(transactionHash,6)='0x68b7';

 Planning Time: 0.197 ms
 JIT:
   Functions: 20
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 1.073 ms, Inlining 0.000 ms, Optimization 0.000 ms, Emission 0.000 ms, Total 1.073 ms
 Execution Time: 5.187 ms
```

```
explain  analyse select * from (select * from transactions where left(transactionHash,6)='0x68b7') as alias  where transactionHash='0x68b71d202dc52ad80812b563f3f6b0aaf1f19c04c1260d13055daad5b88a36a8';

 Planning Time: 0.231 ms
 JIT:
   Functions: 38
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 2.291 ms, Inlining 3.261 ms, Optimization 184.199 ms, Emission 116.861 ms, Total 306.612 ms
 Execution Time: 310.527 ms
```

transfers

1630485594 rows

区块	大小	大小
1-100W	5555691	1593M 1950M
100W-200W	16466034	4871M 5931M
200W-300W	89583794	23G 29G
300W-400W	53202771	15G 19G
400W-500W	288270628	82G 100G
500W-600W	346346564	97G 119G
600W-700W	311496678	86G 106G
700W-800W	341815644	94G 116G
800W-850W	177747790	49G 60G

blockNumber

```
explain analyse select * from transfers where blockNumber=6810068;

Planning Time: 0.935 ms
Execution Time: 0.538 ms
```

transactionHash

```
 explain analyse select * from transfers where left(transactionHash,6)='0x68b7';

  Planning Time: 0.305 ms
 JIT:
   Functions: 20
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 2.178 ms, Inlining 0.000 ms, Optimization 0.000 ms,
Emission 0.000 ms, Total 2.178 ms
 Execution Time: 2.781 ms
```

```
expalin analyse (select * from (select * from transfers where
left(transactionHash,6)='0x68b7') as alias  where
transactionHash='0x68b71d202dc52ad80812b563f3f6b0aaf1f19c04c1260d13055daad5b88a3
6a8';);

 Planning Time: 20.380 ms
 JIT:
   Functions: 38
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 91.872 ms, Inlining 0.000 ms, Optimization 0.000 ms,
Emission 0.000 ms, Total 91.872 ms
 Execution Time: 99.101 ms
```

traces

1255905809 rows

区块	行数	大小
1-100W	2426403	389M 493M
100W-200W	3790413	532M 694M
200W-300W	3727040	534M 694M
300W-400W	18836583	2134M 2943M
400W-500W	147536335	16G 22G
500W-600W	279718500	29G 41G
600W-700W	298750762	31G 43G
700W-800W	331847919	34G 48G
800W-850W	169271854	17G 24G

txHash

```
explain analyse select * from traces where left(txHash,6)='68b71';

 Planning Time: 4.890 ms
 JIT:
   Functions: 60
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 3.734 ms, Inlining 0.000 ms, Optimization 0.000 ms,
Emission 0.000 ms, Total 3.734 ms
 Execution Time: 52.544 ms
```

events

426066340 rows

transactionHash

```
explain analyse select * from events where left(transactionHash,6)='0x68b8';

 Planning Time: 7.163 ms
 JIT:
   Functions: 20
   Options: Inlining true, Optimization true, Expressions true, Deforming true
   Timing: Generation 1.316 ms, Inlining 0.000 ms, Optimization 0.000 ms,
Emission 0.000 ms, Total 1.316 ms
 Execution Time: 539.952 ms
```

区块	行数	大小
1-100W	747943	576M 609M
100W-200W	1373224	862M 922M
200W-300W	2226740	1173M 1269M
300W-400W	8904415	4665M 5048M
400W-500W	58090102	32G 34G
500W-600W	105106164	55G 59G
600W-700W	96480847	52G 56G
700W-800W	98146938	52G 56G
800W-850W	54989967	30G 32G

索引优化未来工作

1. 裁剪索引尺寸。因为上述5个表中每个表的常用查询维度比较相同，可以尝试根据表与表之间的关系，去除重复索引项。从而达到缩减索引的效果。
2. 加快索引速度。比如由于traces表很大，直接用txHash可能查找很慢，可以尝试使用txHash到transactions表中查找到tx对应的blockNumber，然后再traces表中使用blockNumber来查找对应的txHash值。