

Improved Hashtables

```
SELECT SUM(I_extendedprice)
FROM lineitem
WHERE I_shipdate >= '1995-01-01'::date
      AND I_shipdate <= '1996-12-31'::date;
```

Improved Hashtables

```
SELECT I_partkey, SUM(I_extendedprice)
FROM lineitem
GROUP BY I_partkey
ORDER BY SUM(I_extendedprice) DESC
LIMIT 10;
```

Improved Hashtables

Limit (actual time=..18529.047 rows=10)

-> Sort (actual time=..18529.045 rows=10)

Sort Key: (sum(l_extendedprice)) DESC

Sort Method: top-N heapsort Memory: 25kB

-> HashAggregate (actual time=..18427.389 rows=1000000)

Group Key: l_partkey

-> Seq Scan on lineitem (actual time=..3106.676 rows=29999795)

Execution time: 18536.564 ms

Improved Hashtables

Limit (actual time=..14087.096 rows=10)

-> Sort (actual time=..14087.093 rows=10)

Sort Key: (sum(l_extendedprice)) DESC

Sort Method: top-N heapsort Memory: 25kB

-> HashAggregate (actual time=..14047.655 rows=1000000)

Group Key: l_partkey

-> Seq Scan on lineitem (actual time=..3136.398 rows=29999795)

Execution time: 14102.279 ms

Improved Hashtables

Aggregate (actual time=...6691.528 rows=1)

-> Bitmap Heap Scan on lineitem (actual time=..5195.704 rows=9113219)

Recheck Cond: ((l_shipdate >= '1995-01-01'::date) AND (l_shipdate <= '1996-12-31'::date))

Heap Blocks: exact=585542

-> Bitmap Index Scan on i_l_shipdate (actual time=..4429.997 rows=9113219)

Index Cond: ((l_shipdate >= '1995-01-01'::date) AND (l_shipdate <= '1996-12-31'::date))

Execution time: 5196.203 ms

Improved Hashtables

Aggregate (actual time=..3341.864 rows=1)

-> Bitmap Heap Scan on lineitem (actual time=..2569.560 rows=9113219)

Recheck Cond: ((l_shipdate >= '1995-01-01'::date) AND (l_shipdate <= '1996-12-31'::date))

Heap Blocks: exact=585542

-> Bitmap Index Scan on i_l_shipdate (actual time=..951.016 rows=9113219)

Index Cond: ((l_shipdate >= '1995-01-01'::date) AND (l_shipdate <= '1996-12-31'::date))

Execution time: 3343.101 ms

Faster Expression Evaluation

```
SELECT count(*)  
FROM lineitem  
WHERE l_extendedprice > 1000;
```

Faster Expression Evaluation

ExecQual()

FunctionCallInvoke(ExecMakeFunctionResultNoSets)

check_stack_depth()

FunctionCallInvoke(ExecEvalFuncArgs)

FunctionCallInvoke(ExecEvalScalarVarFast)

FunctionCallInvoke(ExecEvalConst)

FunctionCallInvoke(float8gt)

Changing Executor Tuple Flow

```
SELECT I_partkey, SUM(I_extendedprice)
FROM lineitem
GROUP BY I_partkey
ORDER BY SUM(I_extendedprice) DESC
LIMIT 10;
```

Limit (cost=1071965.46..1071965.49 rows=10)

-> Sort (cost=1071965.46..1074578.60 rows=1045255)

Sort Key: (sum(I_extendedprice)) DESC

-> HashAggregate (cost=1038925.33..1049377.88 rows=1045255)

Group Key: I_partkey

-> Seq Scan on lineitem (cost=0.00..888926.22 rows=29999822)