Graph Processing on an “almost” relational database

Ramesh Subramonian
Oracle Labs[1]
ramesh.subramonian@oracle.com

[1] Work done while at LinkedIn

Context

- Not responsibe for recovery/backup/…
- Data sets needed for specific problem small enough for 1 machine
- No need for fault tolerance
- Analyses performed in reflective mode, not reactive mode
- Problem definition is changing rapidly

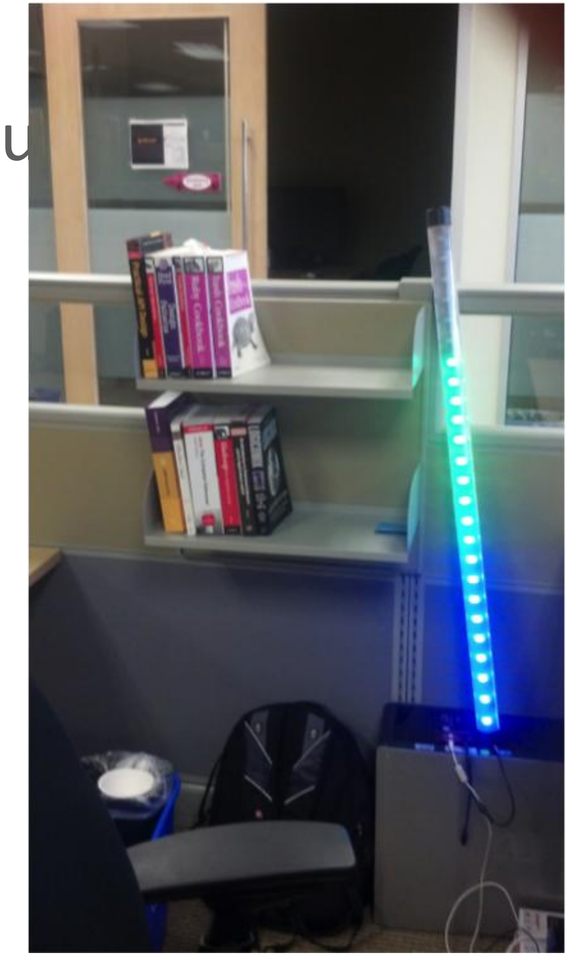
Some Sample Problems

1. Second Degree Network
2. Incremental Path Navigation
3. Filtered Endorsements
4. People You Should Know (as opposed to PYMK)

Inspiration

Motivation

- We must not think of the things we could do with, but only of the things that we can’t do without.
- Let your boat of life be light, packed with only what you
- You will have time to think as well as to work.



Three Men in a Boat, Jerome K. Jerome

Motivation

- Tables are at a lower level of abstraction than relations
  - they give the impression that positional (array-type) addressing is applicable (which is not true of n-ary relations)
  - they fail to show that the information content of a table is independent of row order
- Nevertheless, even with these minor flaws, **tables are the most important conceptual representation of relations**, because they are universally understood

Codd, Turing Award Lecture

Inspiration

- Ease of expressing constructs arising in problems
- Suggestivity
- Ability to subordinate detail
- Economy
- Amenability to formal proofs

- *Debugging Support – test as you go*

Iverson, Turing Award Lecture

Second Degree Network – Data Structure

Members (Nodes)

index	Member ID (mid)	TC_{lb}	TC_{ub}
0	100	0	2
1	200	2	3
2	300	…	…

Connections (Edges)

index	from	to	to_idx
0	100	200	1
1	100	300	2
2	200	100	0

Bad programmers worry about code. Good programmers worry about data structures

ORACLE

Second Degree Network – Algorithm

- Find *i* such that $T_M[i].mid = m$ (fast because T_M is sorted on mid)
- i=`q f_to_s TM mid "op=[get_idx]:val=[$m]”
- Find range of rows in T_C that contain edges out of *m*
- TC_lb=`q f_to_s TM TC_lb "op=[get_val]:idx=[$i]"`
- TC_ub=`q f_to_s TM TC_ub "op=[get_val]:idx=[$i]“`
- Create temp table TD_1 by copying column to_{idx} for above rows
- q copy_fld_ranges TC to_idx “” $TC_lb $TC_ub TD1

ORACLE

Second Degree Network – Algorithm (contd)

- Repeat previous step for each row of TD_1 to create TD_2
  – By using field to_{idx} and not field *to*, we avoid searching T_M for each entry of TD_1
- Implemented as ``user-defined function''
- De-dupe members in TD_2 to create output T_{out}
- q mk_uq TD2 mid Tout mid

ORACLE

Performance Numbers – Second Degree Network

Size (1st Degree)	Size (2nd Degree)	Time (msec)
120	64349	8.070
263	112213	12.53
505	334246	41.51
1021	694644	80.13
2053	1166594	166.4
4091	1956817	259.4
8199	4069339	1363
16378	8319301	1516