

Billionth Car

Design a system such that 100 billionth visitor on Google, gets a prize in raffle

i.e. when you think about it, you realize that they are looking for a good method of counting in a distributed manner.

Distributed counting is a hard problem. Not hard as in not solvable, but hard as in "I didn't realize it can become so complicated just to count!" hard.

Reading pointers:

- Canonical answer (CRDT): <http://www.cakesolutions.net/teamblogs/how-to-build-a-distributed-counter>
- Reasonably acceptable answer (a bit hand-wavy, but conveys that you know enough): See 'Sharded Counters' in this article <http://highscalability.com/blog/2009/2/18/numbers-everyone-should-know.html>
- Lazy answer: Use a Distributed KV Store that does this: <http://blog.memsql.com/high-speed-counters/>

In practical systems, this will be mixed with existing data pipelines:

- How FB does it: <http://highscalability.com/blog/2011/3/22/facebooks-new-realtime-analytics-system-hbase-to-process-20.html>
- How Twitter does it: <https://www.slideshare.net/kevinweil/rainbird-realtime-analytics-at-twitter-strata-2011/40-Multiple-Formulas-So-far-we>

(For the adventurous) Research paper: <http://arxiv.org/pdf/1307.3207v1.pdf>

