Parameterizing Locality and Coordination

Marc Brooker

Amazon Web Services

What’s the Problem

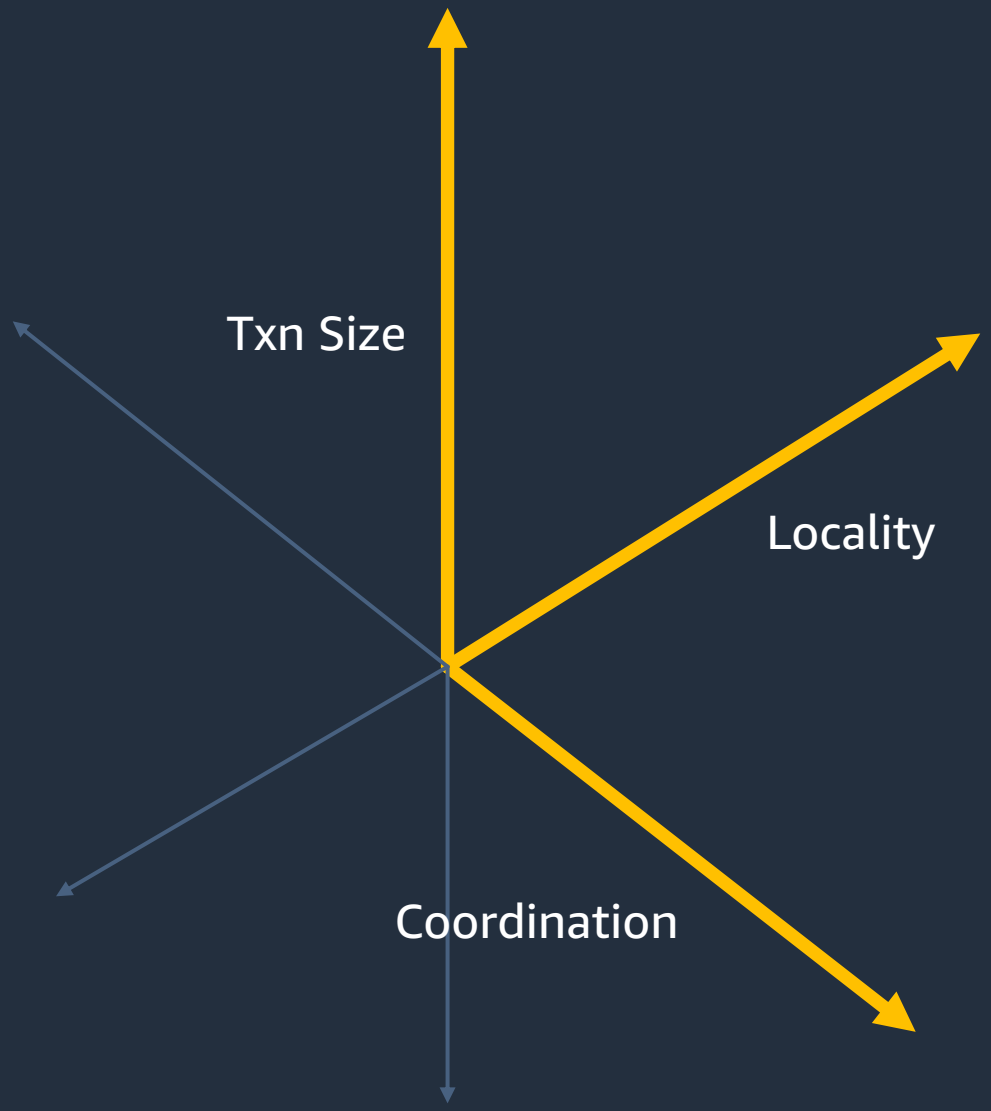
- Distributed transaction processing is hard
- Performance depends on workload!

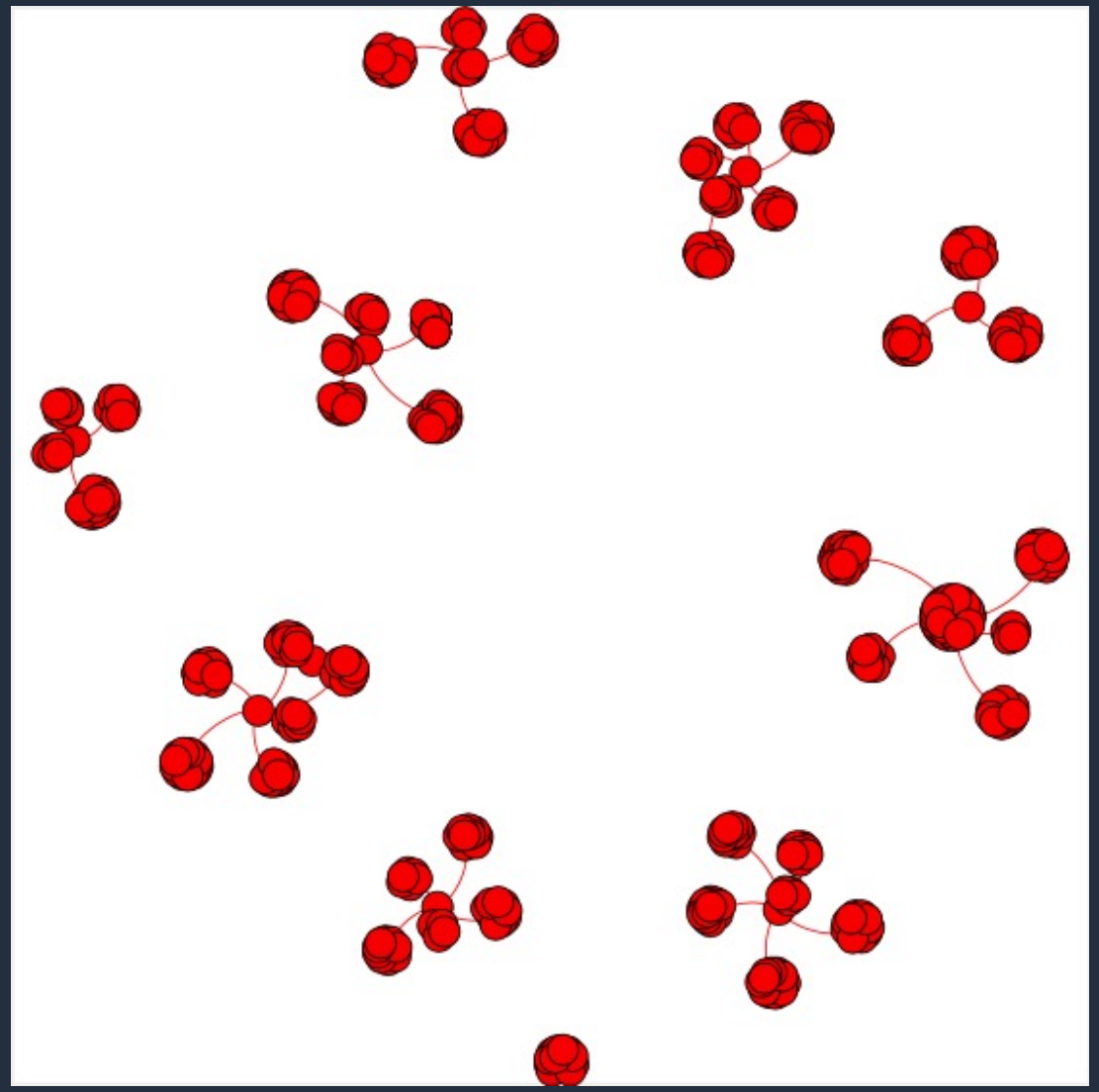
What's the Problem

- Locality and Coordination are critical properties of workloads
- Locality
  - Local: Caching
  - Distributed: How many machines I need to hit
- Coordination
  - Concurrency Control: What blocks, what aborts, etc
  - Commit: How many machines I need to hit

Today’s Benchmarks Don’t Help

- YCSB, TPC-C are just too easy
- TPC-E is ok, but only a point sample
  - And still pretty easy
- Real workloads have a lot more diversity
- Many are *harder*





Graphs and Workloads

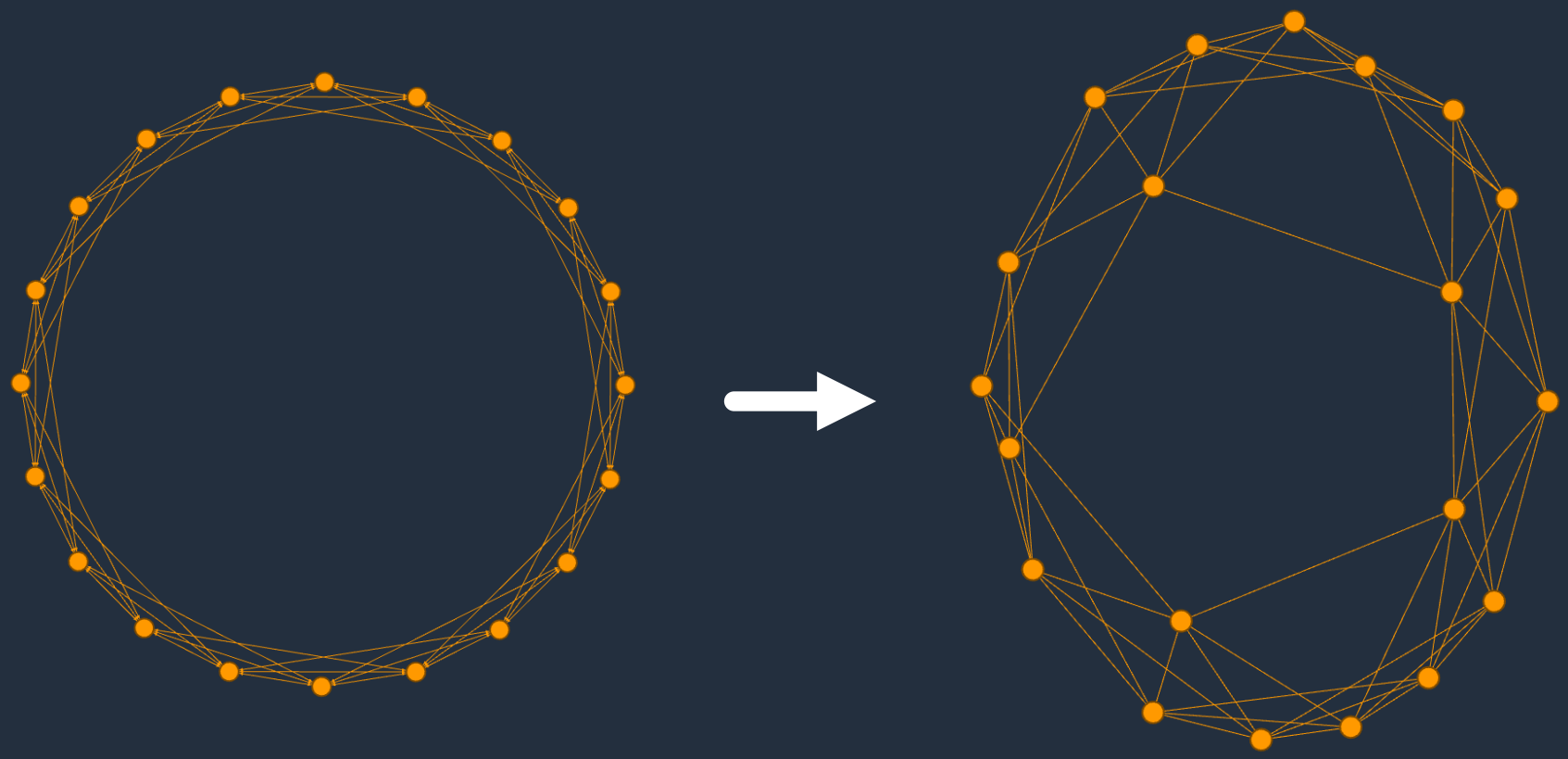
- Locality relates to *degree distribution* and *clustering*
- Coordination relates to *path length* and *degree distribution*

- Generate a graph, walk it => workload!

Generating Graphs: Watts-Strogatz

- Watts-Strogatz *Small World Networks* model
  - Generation is O(N)
  - Single parameter: controls connectedness and locality

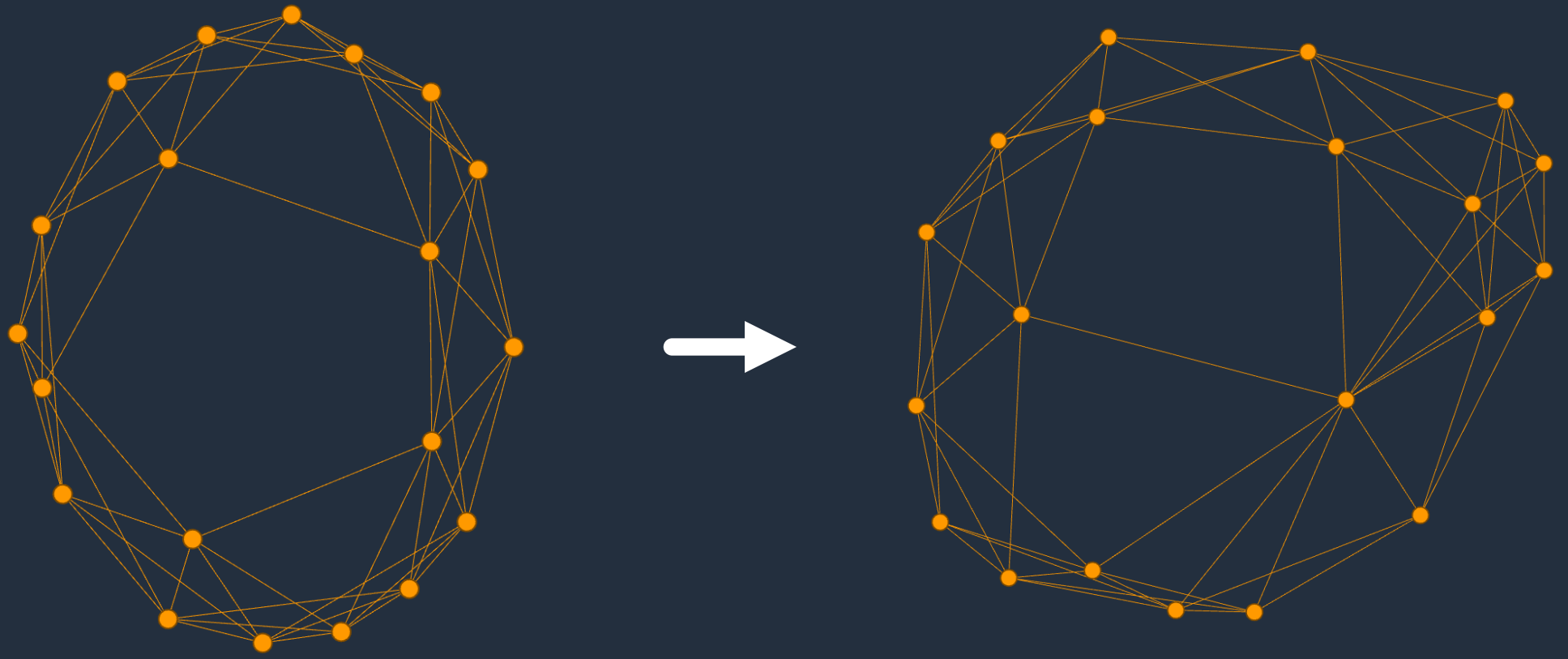
Watts and Strogatz, *Collective dynamics of 'small-world' networks*, Nature, 1998



Generating Graphs: Watts-Strogatz-Zipf

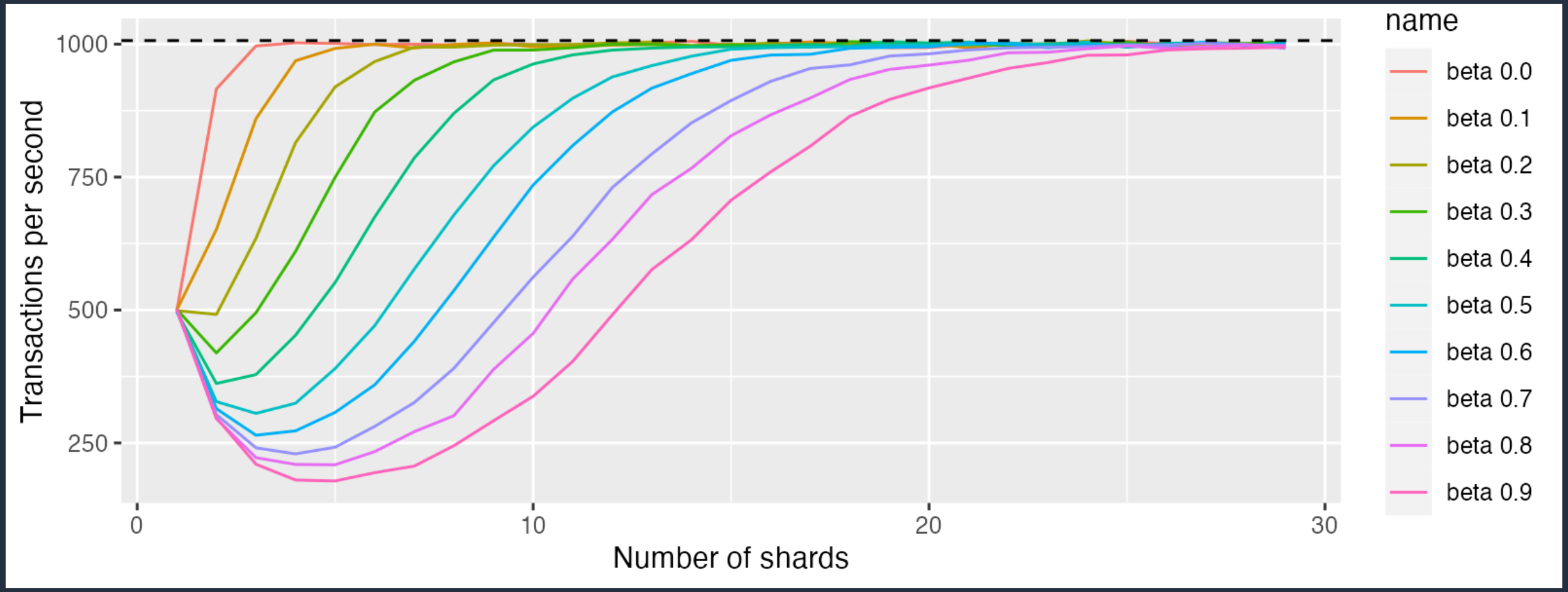
- Database degree distributions tend to be more extreme
- “Hot Keys”
- So extend W-S model with non-uniform (e.g. Zipf)

- Two parameters now:
  - W-S *p*, controls connectivity and locality
  - Zipf *alpha,* controls key heat distribution



Extensions

- Depending on Isolation Level
  - Write-Write edges only (e.g. Snapshot)
  - Read-Write edges (e.g. Serializable, ANSI RR)
  - *Coloring* graph edges, or separate R and W graphs
- Hierarchical models to capture known structure
- Other network models



Thank you!

Marc Brooker

mbrooker@amazon.com

marcbrooker@gmail.com

https://brooker.co.za/blog/