Scalable Causal Consistency for Wide-Area Storage with COPS
Wyatt Lloyd (Princeton), Michael J. Freedman (Princeton), Michael Kaminsky (Intel Labs), David G. Andersen (CMU)

Motivation
- Distributed data stores support complex online applications
  - e.g. social networks
- Theory constrains properties
  - CAP Theorem
  - Seq Consistency || Low Latency
- Most practical systems adopt eventual consistency
  - Complicates application logic
  - Exposes inconsistencies to users

Causal+ Consistency
- **Causal** consistency
  - Related ops appear in the correct order
- **Plus** convergent conflict handling
  - Conflicting puts are handled identically in each DC
- Spectrum of Consistency Models:
  - Linearizability > Seq. > Causal+ > FIFO
  - PK Seq > Eventual

Causal+ Examples
1) Alice uploads photo
2) Alice adds photo to album
A) Carol sets coffee.time = 8am
B) Dave sets coffee.time = 10am
Causal+: Referential integrity. Photo always exists before album.
Eventual: Broken reference in album is possible.
Causal: Forever divergent times are possible.

Clustering of Order Preserving Servers

Challenges
- Minimize space footprint
- Garbage collect old state
- Minimize overhead of consistent replication
- Leverage transitivity of causality
- Ensure fast get transactions:
  - Worst-case 2 rounds under concurrent writes
  - Get_by_version

Implementation
- Built on top of FAWN-KV
- ~13,000 LOC
- Latency < 1ms
- Throughput similar to weaker systems
- Scales linearly