



**NETFLIX**



Ephemeral **V**olatile Cache

Distributed  
Memcached

Tunable  
Replication

High  
Resilience

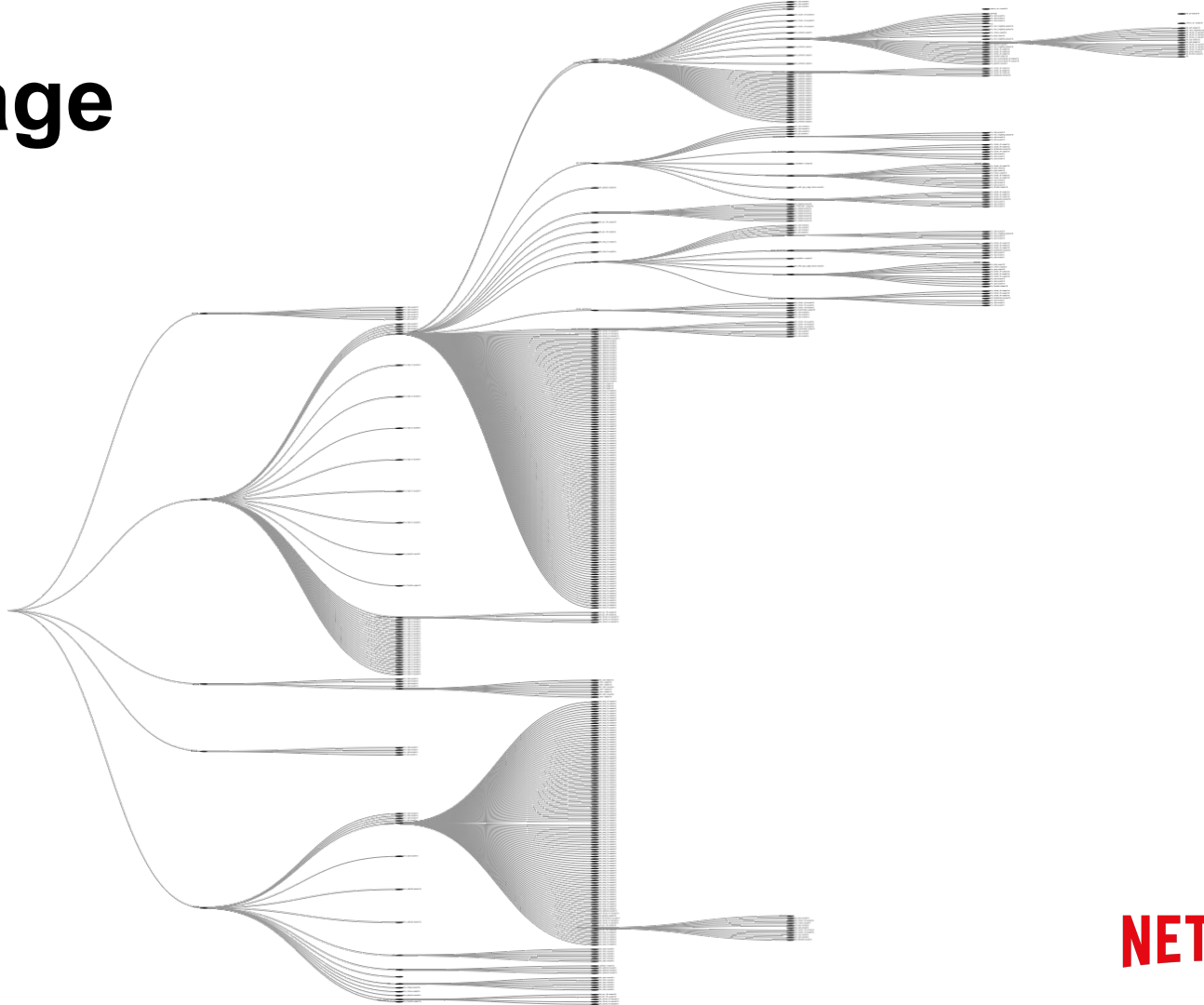
Topology  
Aware

Data  
Chunking

Additional  
Functionality

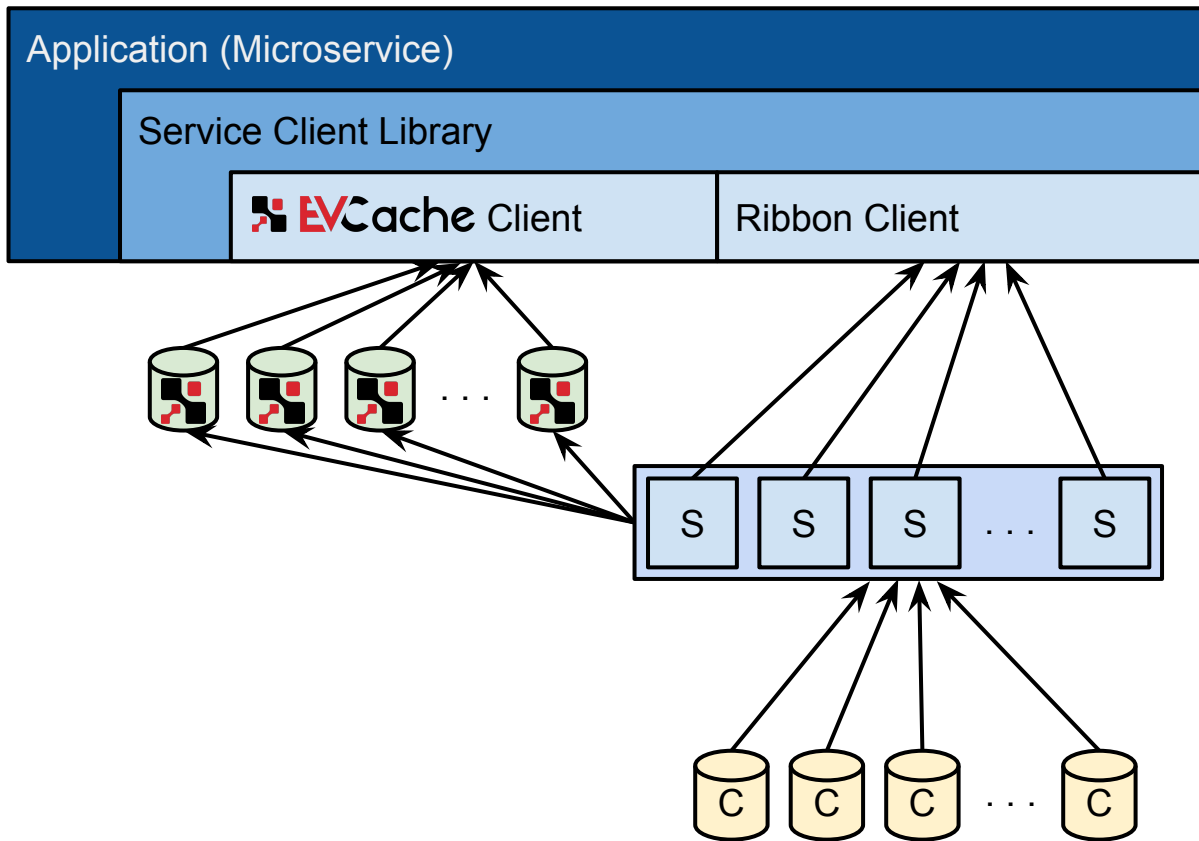
**NETFLIX**

# Home Page Request

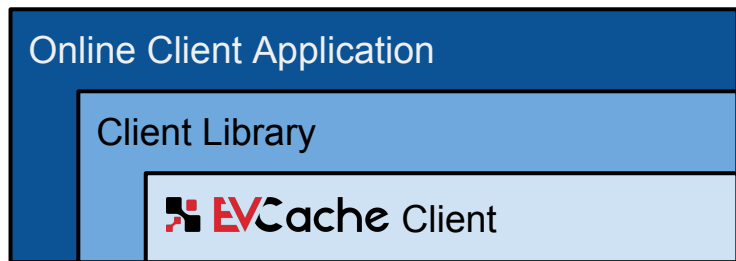


**NETFLIX**

# Use Case: Lookaside cache

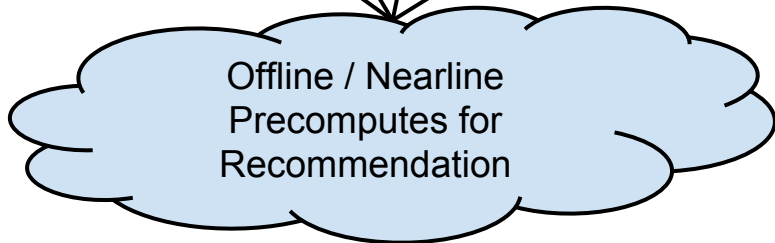
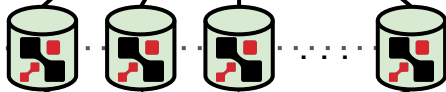


# Use Case: Primary Store



Online Services

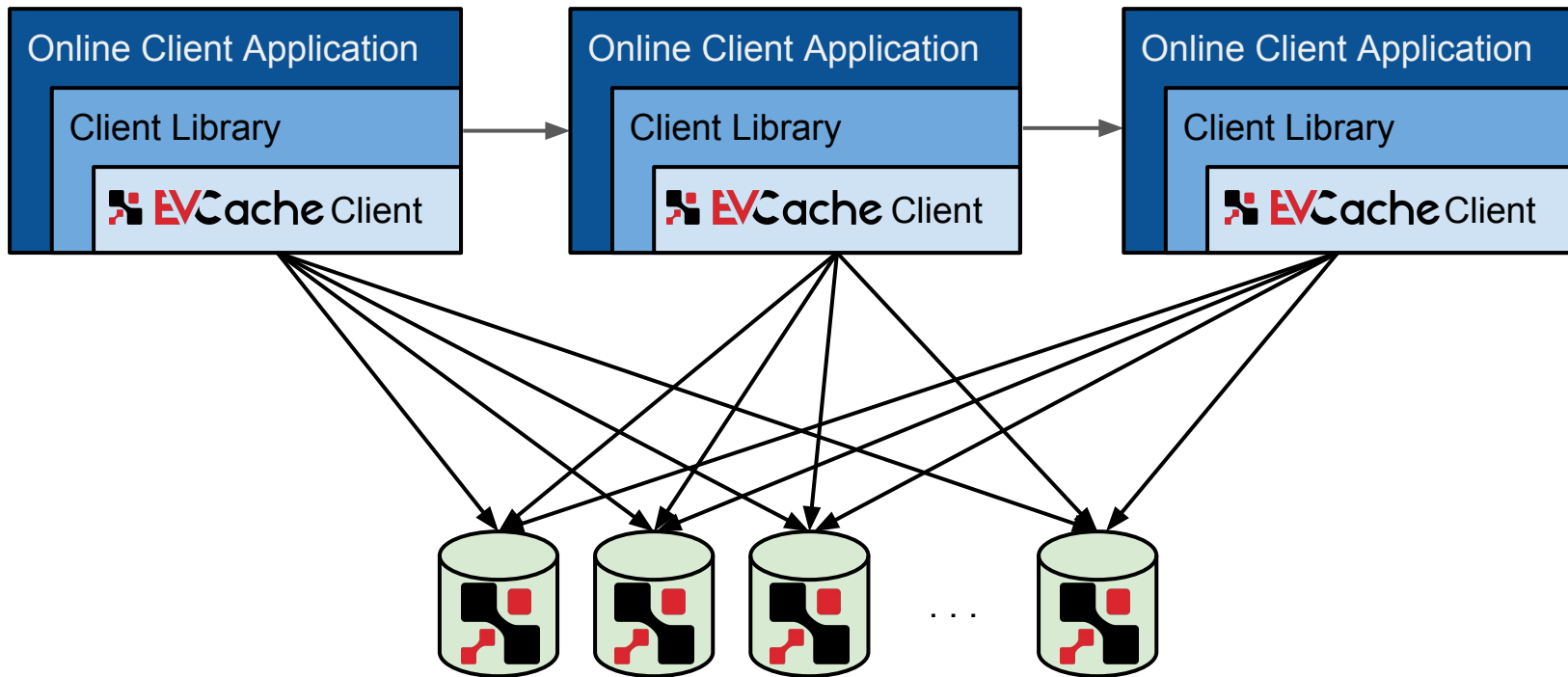
Offline Services



Data Flow →

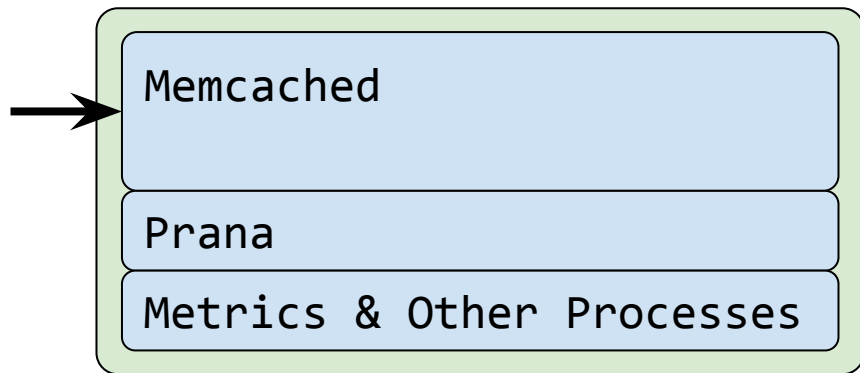
**NETFLIX**

# Use Case: Transient Data Store



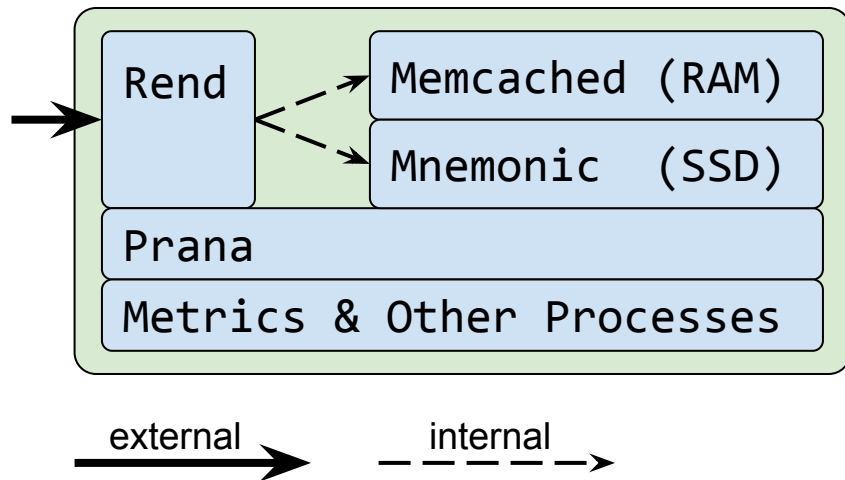
# Current Server

- All data stored in RAM (Memcached)
- Became more expensive with expansion / N+1 architecture



# New Server

- Adds Rend and Mnemonic
- Still looks like Memcached
- Unlocks cost-efficient storage & server-side intelligence





# Rend

- High-performance Memcached proxy & server
- Written in Go
  - Powerful concurrency primitives
  - Productive and runs fast
- Manages the L1/L2 relationship
- Server-side data chunking
- Tens of thousands of connections

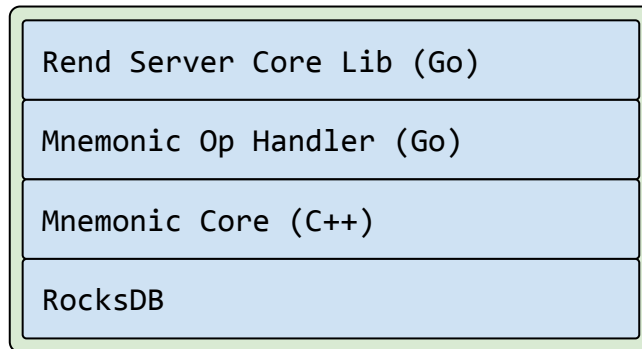
Connection Management		M E T R I C S
Server Loop	Protocol	
Request Orchestration		
Backend Handlers		

NETFLIX

# Mnemonic

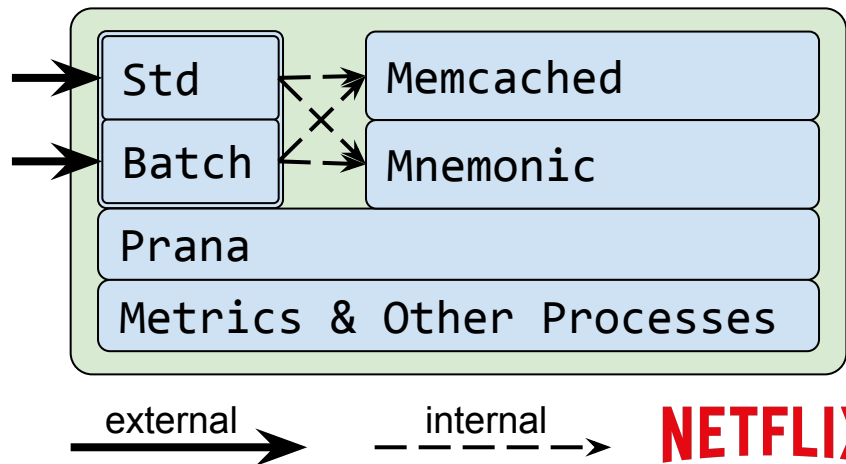
- Manages data storage to SSD
- Reuses Rend server libraries
  - Handles Memcached protocol
- Mnemonic core logic
  - Implements Memcached operations into RocksDB
  - Includes sharding to bound latency

## Mnemonic Stack



# Moneta in Production

- Serving some of our most important personalization data
- Two ports
  - One for regular users (read heavy or active management)
  - Another for "batch" uses: Replication and Precompute
- Maintains working set in RAM
- Optimized for precomputes
  - Smartly replaces data in L1



# Open Source

<https://github.com/netflix/EVCache>

<https://github.com/netflix/rend>

# Contact Info

[@sgmansfield](#)

[smansfield@netflix.com](mailto:smansfield@netflix.com)

[@netflix](#)

<http://techblog.netflix.com/>