Benchmark Architecture - Overhead Measurement Strategy

Pure Workloads (Baseline)

Benchmark: View (BenchmarkECDSASign, etc.)
Measures: Raw workload TPS without any overhead

Workload Characteristics:

ecdsa: CPU + heavy allocations cpu: Pure CPU cycles

noop (minimal) | cpu (CPU burn) | ecdsa (CPU + alloc) | echo (I/O)

Add gRPC Stack

Workloads + gRPC Stack

Benchmarks: BenchmarkGRPCBaseline, BenchmarkGRPCImpl, BenchmarkRemote
Measures: gRPC overhead (serialization, RPC, network)

Local (Baseline gRPC) Go microbenchmark | Local (FSC gRPC Impl) Go microbenchmark | Remote (Client-Server) Network overhead

All workloads (noop, cpu, ecdsa, echo) invoked via gRPC client

Compare: Baseline vs FSC Impl
Measure: gRPC implementation overhead

Compare: Local vs Remote
Measure: Network overhead

Add FSC Node Runtime

Workloads + gRPC + FSC Node Runtime

Benchmarks: BenchmarkAPI, BenchmarkAPIGRPC, BenchmarkAPIGRPCRemote
Measures: FSC node runtime overhead (view execution, session mgmt, identity)

Direct View API (No view factory) Go microbenchmark | gRPC View API (Local) (with view factory) Go microbenchmark | gRPC View API (Remote) (Client-Server) Network overhead

All workloads executed through FSC node runtime (view runtime)

Compare: API vs gRPC
Measure: gRPC overhead in FSC context

Compare: Local vs Remote
Measure: Network + FSC overhead