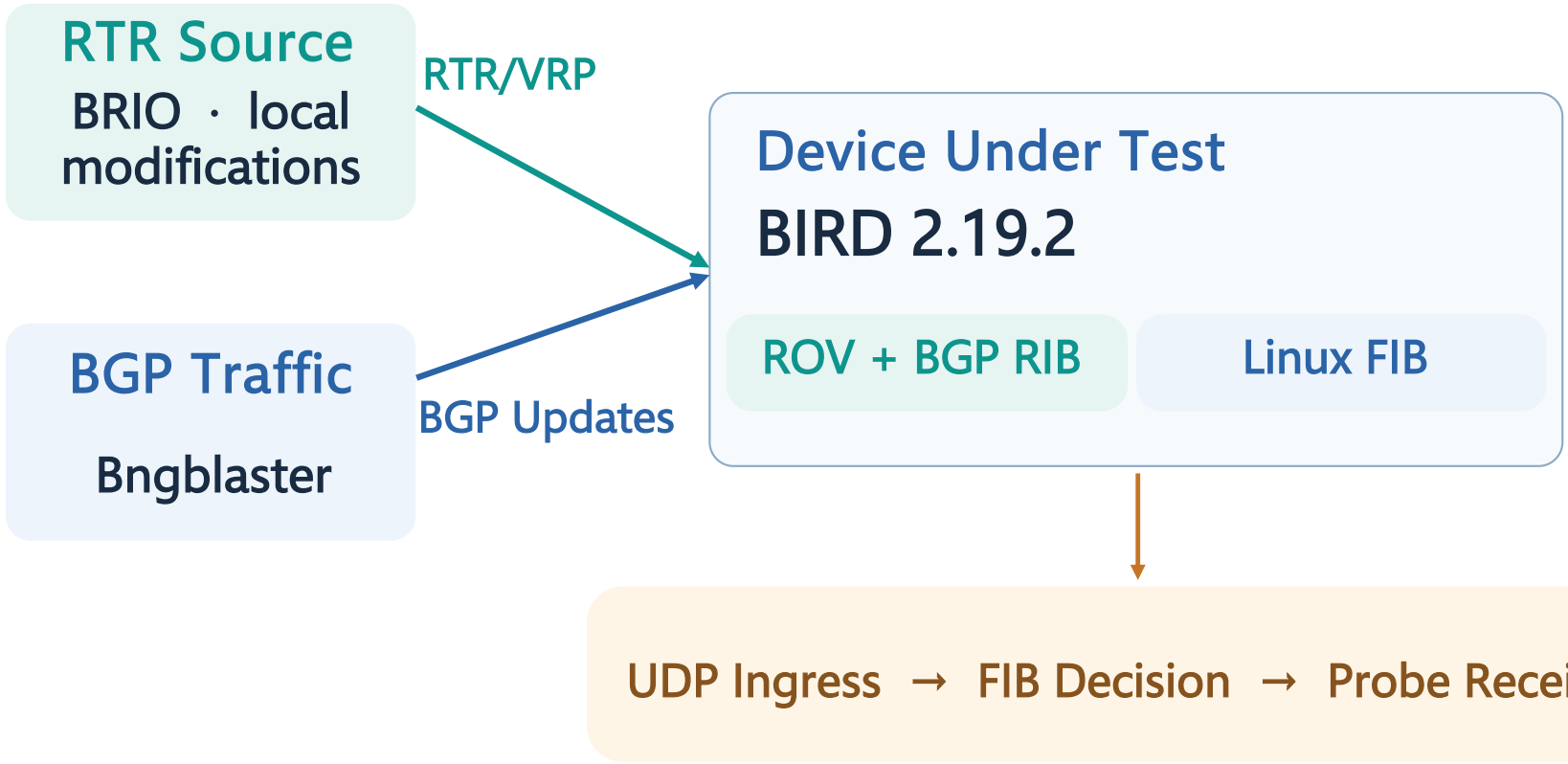
The **Updates** on the Benchmarking Methodology for Route Origin Validation (ROV)

draft-lg-bpm-benchmarking-methodology-for-rov-00

Presenter: Libin Liu

IETF 127 BPM WG, San Francisco, USA

Implemented a Repeatable ROV Benchmark Testbed



Instrument Four Observables

VRP install	BIRD VRP logs
ROV state	BIRD CLI queries
RIB updates	Best-path logs
FIB changes	UDP probe receivers

Reproducibility WSL2 Linux · Intel Core 5 220H · 15.37 GiB · modified Bngblaster (IPv6 peers) and BRIO (send timestamps)

Real-world Data as a Controlled Benchmark Baseline

BGP | RIPE RIS

1,221,131 BGP route candidates

999,125 IPv4 · 222,006 IPv6

99 peers · 805,052 distinct prefixes · 9 Sep 2026

VRPs | RPKIViews

1,060,368 processed VRPs

820,699 IPv4 · 239,669 IPv6

8 Sep 2026 · Test ROAs added

Four Controlled Workload Families

Single-batch VRP	50k–1M baseline; 1,000 VRP operations
Selected-prefix ROV	100/500/1,000 prefixes; local/distributed
Sustained VRP churn	500k base; 300 batches; 100/500/1,000 ops
BGP-triggered events	1,000 BGP operations; ROV enabled/disabled

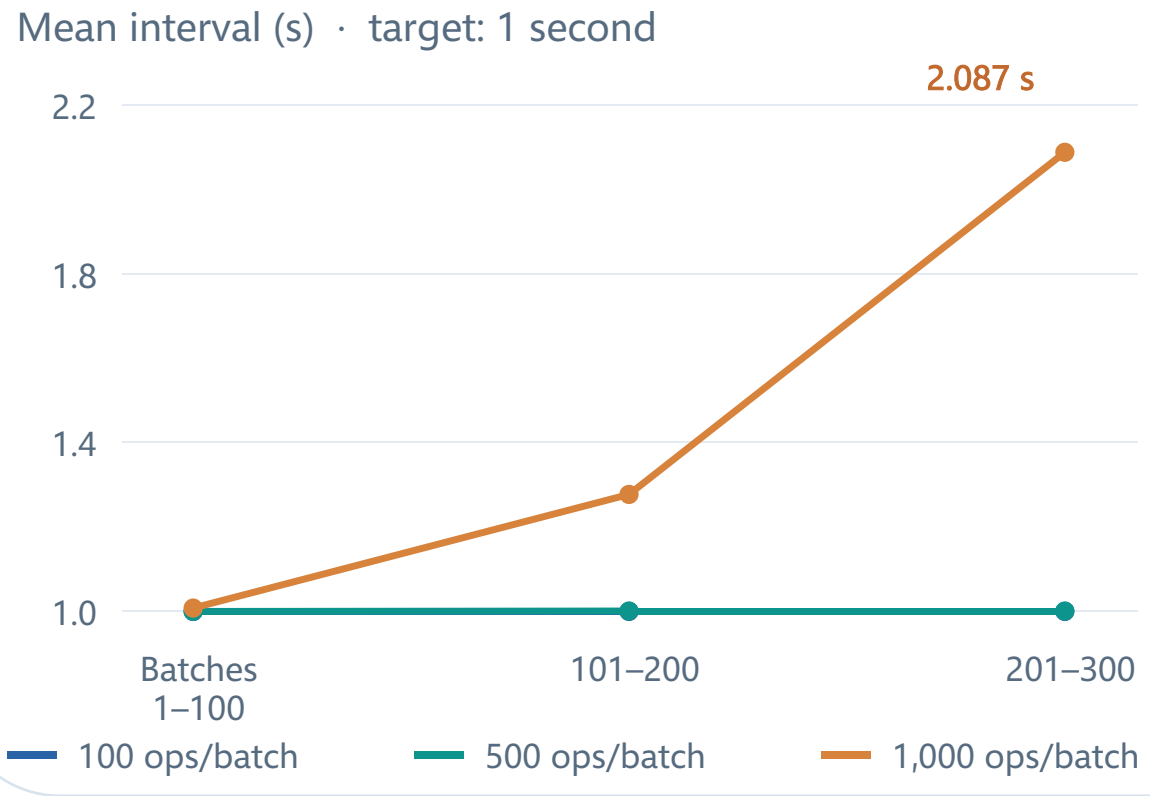
Benchmark Procedures Implemented and Evaluated

Benchmark Test	Controlled Procedure	Recorded Measurements
VRP update App. B.5	Incremental vs. full resync; 50k–1M VRPs	VRP installation latency
ROV validation App. B.6	Local vs. distributed; 100/500/1,000 selected prefixes	Post-installation latency; routes actually affected
BGP convergence App. B.7	VRP-triggered and BGP-triggered, ROV on/off	E2E and post-install timers; RIB/FIB changes
Scalability & churn App. B.8–B.9	50k–1M VRPs; 300 batches; 100–1,000 ops/batch	Latency distributions; delivery intervals; stability
Resource utilization App. B.10	Idle, full sync, and sustained updates	System and BIRD CPU / memory (RSS)
RTR session behavior App. B.11	Session reset, cache reset, malformed PDUs	Recovery, revalidation, correctness

Results in appendix B: latency statistics (including P95), BGP convergence, CPU/RSS, and recovery observations.

Implementation Exposed Measurement and Comparability Issues

Serial Notify Interval During 300-Batch Churn



What testing taught us

01 Timestamp boundaries
Separate RTR delivery, VRP installation, ROV state, and RIB/FIB completion.

02 Observer and generator overhead
CLI polling and probes can delay observation; BRIO sends can drift under churn.

03 Comparable affected workloads
1,000 selected prefixes caused 3,640 vs 23,777 actual state changes.

Methodology lesson: observed delays are not always DUT-only; specify timing, monitoring overhead, and actual affected routes.

Community Feedback and Our Response

- Comments from Adrian Cole
  - Is a dedicated ROV benchmarking methodology necessary?
  - Can existing BGP benchmarking methodologies cover ROV performance evaluation?

  Our response:
  - **Reuse existing methods**: Build on existing BGP benchmarking methodologies wherever applicable, avoiding duplication
  - **Address measurement gaps**: Focus on ROV-specific measurements, including VRP processing, revalidation, scalability, and RTR recovery

- Comments and interests from Nisar Ali
  - Expressed interest in ROV benchmarking and our experimental platform
  - Asked about latency isolation and CPU/memory bottlenecks under high VRP churn

  Our response:
  - Clarified overlapping processing stages
  - Shared CPU/memory measurements from continuous and isolated VRP update tests

Next Steps

- Clarify the motivation, scope, and relationship to existing BGP benchmarks
- Continue to request feedback from BPM and RPKI community
- Consider a WG adoption call?