

Recent uses of Ark for Innovative Measurements

Matthew Luckie (CAIDA)
AIMS-19



This work is partially supported by U.S. National Science Foundation award OAC-2131987 and CNS-2120399

Outline

- Problem: researchers have an infrastructure gap. Unmet need for a sophisticated, wide-scale, topologically diverse infrastructure.
- Spectrum of possible solutions; we'll review these.

Outline

- Problem: researchers have an infrastructure gap. Unmet need for a sophisticated, wide-scale, topologically diverse infrastructure.
- Spectrum of possible solutions; we'll review these.
- Domain Specific Language (DSL) discussion at AIMS May 2023 workshop occurred over multiple days. Idea:
 - expose a set of measurement primitives (e.g., DNS, ping, traceroute, HTTP)
 - users write applications that use these primitives enhanced with their logic

Domain Specific Language: Pros + Cons

- ✓ High-level abstractions for common measurements that have multiple complicated steps
- ✓ Easier for non-programmers
- ✓ Implementation of primitive is probably better than a roll-your-own
- ✓ Harder to do bad things; send bad packets, too many packets, write to local disk, consume CPU
- ✓ Can describe capabilities to VP hosts

Domain Specific Language: Pros + Cons

- ✓ High-level abstractions for common measurements that have multiple complicated steps
- ✓ Easier for non-programmers
- ✓ Implementation of primitive is probably better than a roll-your-own
- ✓ Harder to do bad things; send bad packets, too many packets, write to local disk, consume CPU
- ✓ Can describe capabilities to VP hosts
- DSL still needs to be learned
- Base language (Python, Lua, Perl, Ruby) might not be well-known
- Limited scope / cut down; might make it difficult to do some types of tasks
- Researchers reliant on DSL maintainers exposing useful primitives and logic
- Researchers reliant on DSL maintainers keeping it up to date to work on modern systems, modern protocols

Our Platform Goals

1. **Easy to use.** The protocol stack, implementation, programming, and sysadmin expertise required for measurement should be minimized.

Our Platform Goals

1. **Easy to use.** The protocol stack, implementation, programming, and sysadmin expertise required for measurement should be minimized.
2. **Performant.** The delay between measurement and result should be small, so that researchers can build complex reactive measurements.

Our Platform Goals

1. **Easy to use.** The protocol stack, implementation, programming, and sysadmin expertise required for measurement should be minimized.
2. **Performant.** The delay between measurement and result should be small, so that researchers can build complex reactive measurements.
3. **Interoperable.** The components that we use should be off-the-shelf and easily deployable to maximize avenues of future deployment.

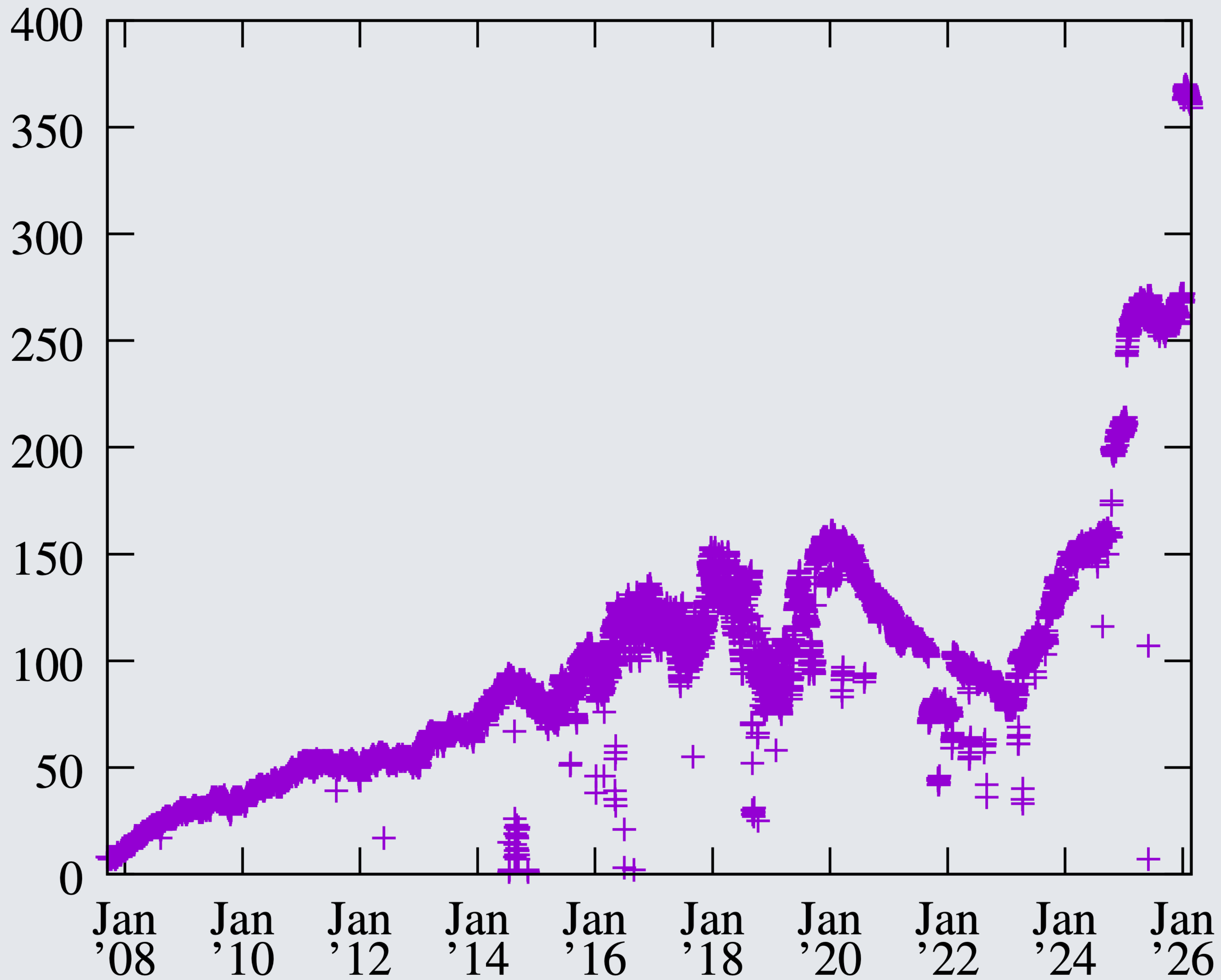
Our Platform Goals

1. **Easy to use.** The protocol stack, implementation, programming, and sysadmin expertise required for measurement should be minimized.
2. **Performant.** The delay between measurement and result should be small, so that researchers can build complex reactive measurements.
3. **Interoperable.** The components that we use should be off-the-shelf and easily deployable to maximize avenues of future deployment.
4. **Site-host transparent.** The environment should allow platform operators to accurately describe the types of measurements the VPs will do.

Recent Supporting Activities

- University of Twente's LaCeS Anycast Analyses
- Characterization of MPLS deployments
- Analysis of DNS ECS deployment
- Presence of Malicious Domains in Recursive Resolvers (Trufflehunter)
- Characterization of QUIC deployments
- Characterization of Video Conferencing performance

Number of Ark VPs



Cloud

Internet2 + NRP

Twente Tangled

Twente Anycast Measurements

- Needs:
 - Lots of VPs for RTT constraints
 - Pings, traceroutes, DNS queries
- Outcomes:
 - Daily public LaCeS anycast census
 - IMC 2025, ANRW 2025, CNSM 2025 papers

Johns Hopkins MPLS measurements

- Needs:
 - traceroutes to uncover tunnel topologies
 - pings to fingerprint routers
- Outcomes:
 - IMC 2025 paper
 - Integration into CAIDA ITDK process

LSU / Twente MudHunter

- Replication / Extension of UCSD TruffleHunter from IMC 2020
- Needs:
 - Lots of VPs to reach different anycast recursive servers
 - DNS queries
- Outcomes:
 - CSET 2025 paper

TU Munich: QUIC fingerprinting

- Needs:
 - Ability to conduct QUIC measurements
 - Vantage points
- Approach:
 - Nikolas Gauder implemented QUIC primitive into scamper
 - We deployed on a subset of Ark VPs to support development
- Challenge:
 - QUIC support appeared in OpenSSL 3.2 — (Debian Trixie)
- In progress: AIMS talk to follow

Colorado / Illinois / Princeton: RT Video

- Needs:
 - Ability to send specific sequences of UDP packets
 - Vantage points
- Approach:
 - Implemented OWAMP to define UDP packet sequences
 - We deployed on a subset of Ark VPs to support development
- In progress: AIMS talk to follow

ITDK

- Needs:
 - Tightly coordinated Vantage points for alias resolution
 - Traceroutes for MPLS-focussed data collection
 - Pings for RTT constraints for geolocation
- Challenges
 - ITDK data collection process takes weeks
 - Graph has become large — ~5M nodes (from ~2M 2 years ago)