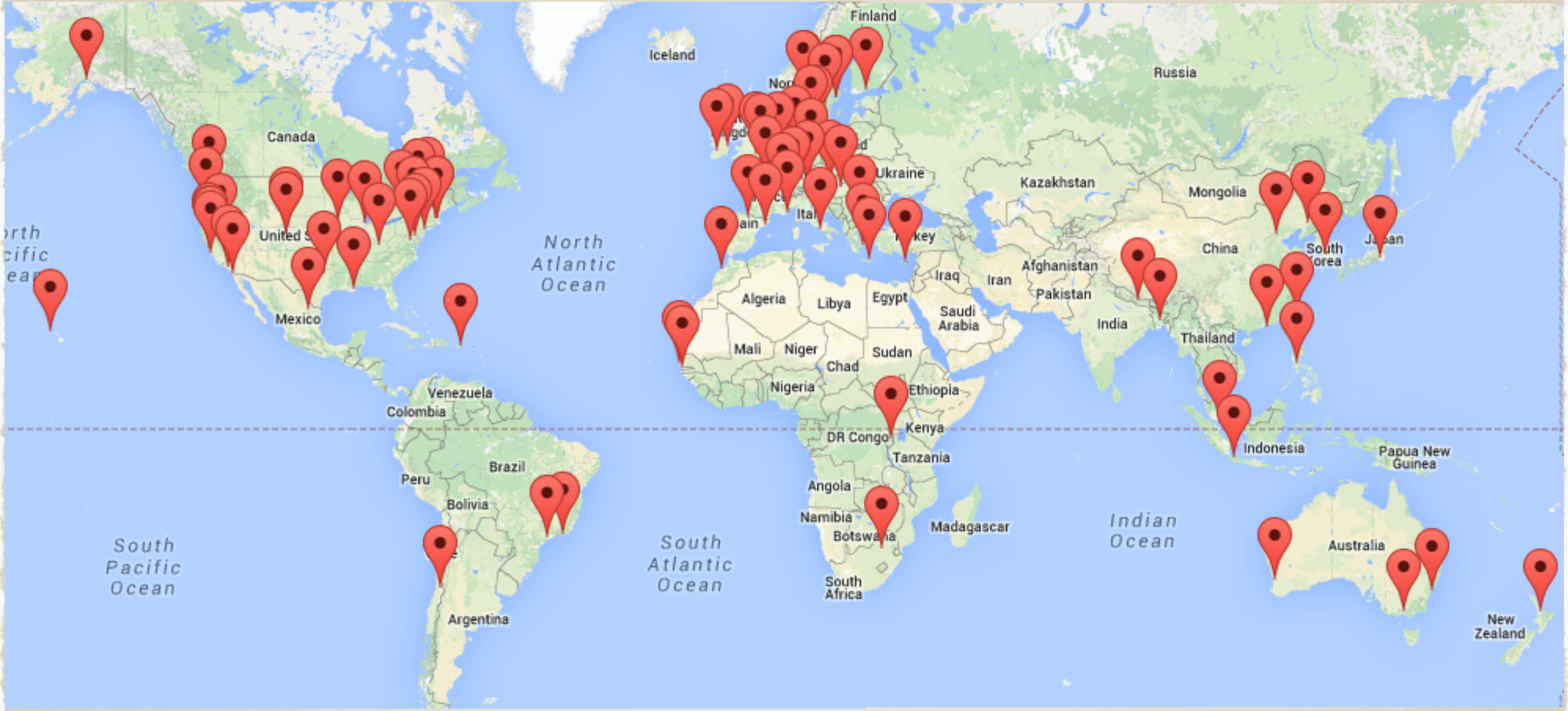
Ark Update: Present & Future

Young Hyun
CAIDA

AIMS 2015 Workshop
Mar 31, 2015

Archipelago
Measurement Infrastructure

Monitor Deployment

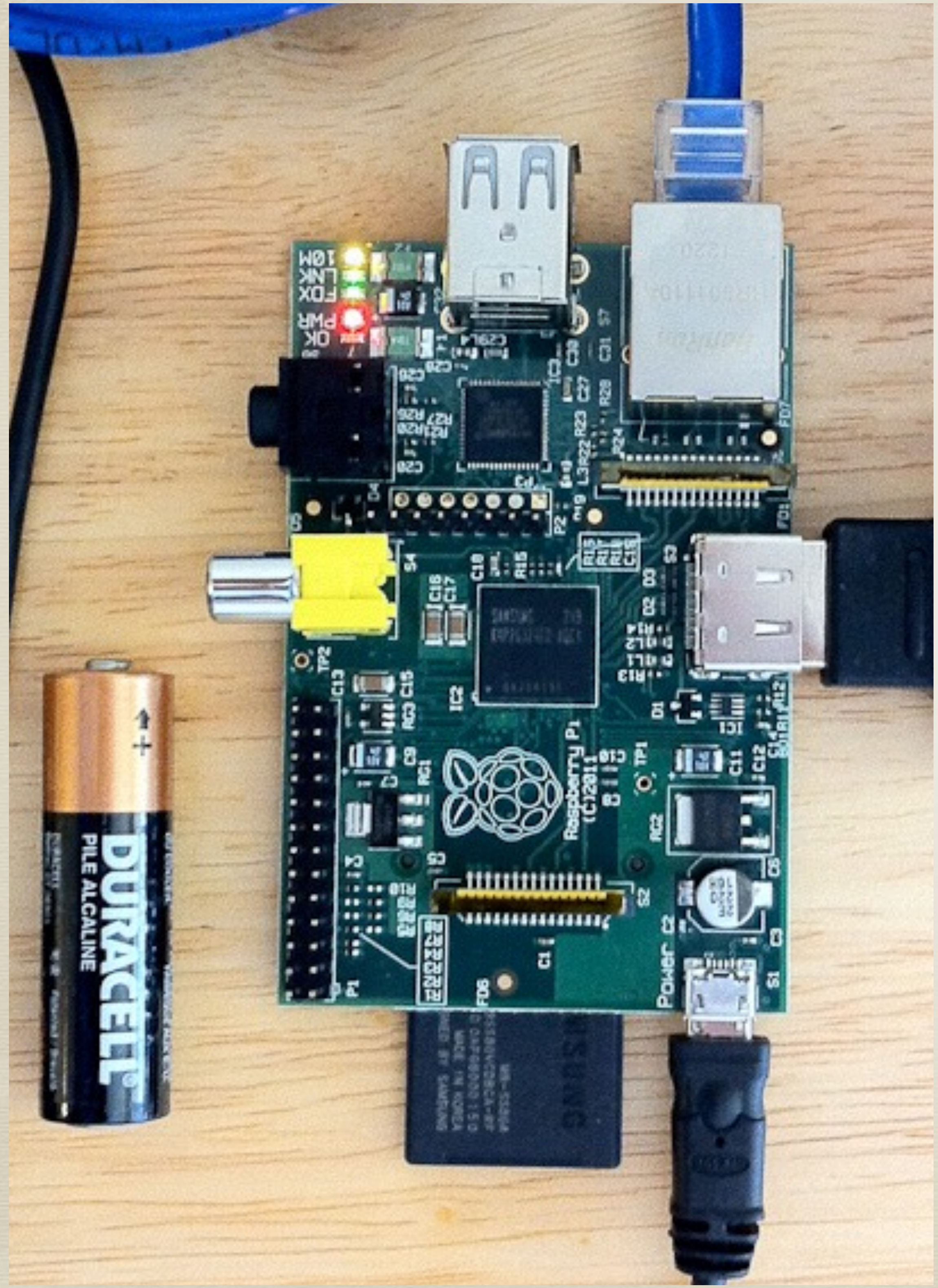


- 107 monitors in 40 countries
  - 59 Raspberry Pi's
  - 44 have IPv6
  - 36 have RADclock

	Continent
39	North America
6	South America
39	Europe
5	Africa
14	Asia
4	Oceania

	Organizations
48	academic
24	residential
23	commercial/business
10	network infrastructure
2	other

Raspberry Pi



1st gen

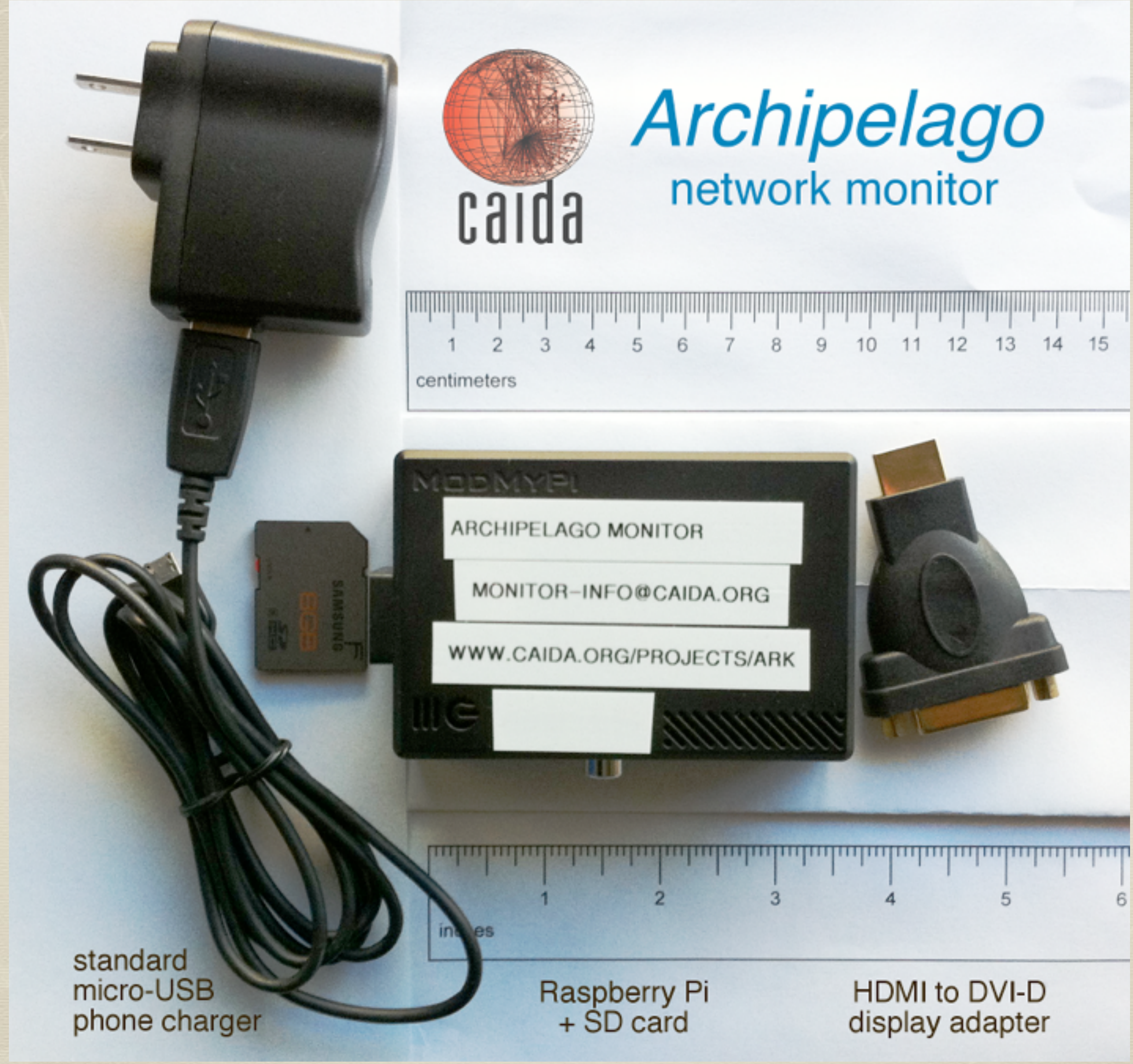
- 700MHz ARMv6
- 512MB RAM

2nd gen

- 900MHz quad-core ARMv7
- 1GB RAM

both

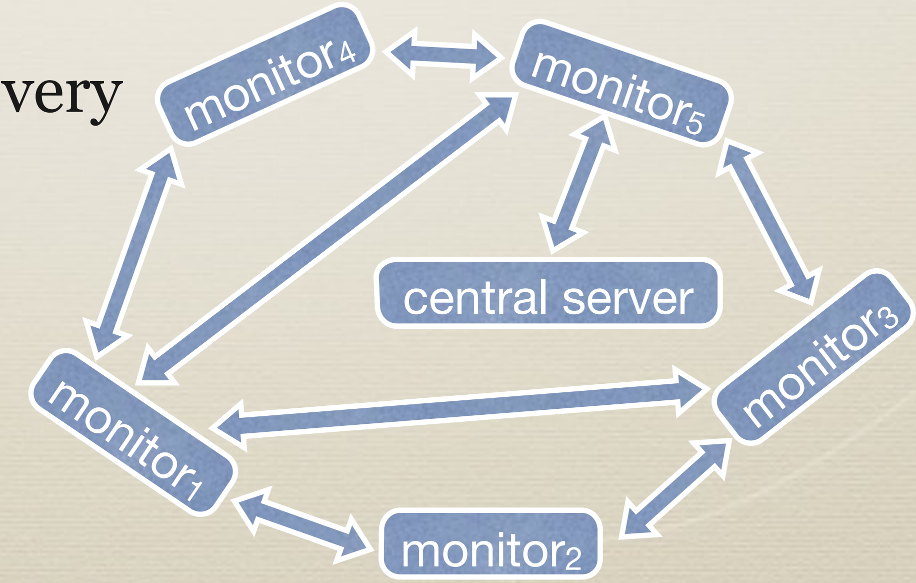
- 100 Mbps Ethernet
- 8GB SD card
- $35 for bare board



~$68 complete system

Tools

- *Marinda* distributed tuple space
  - stores tuples: arrays of strings, numbers, and sub-arrays
  - users retrieve tuples by structural pattern matching (not regex)
  - enables communication and coordination
    - persistent encrypted TCP connections with transparent reconnects
    - decentralized (peer-to-peer) or client-server communication
    - supports broadcast, RPC, publish-subscribe, Bag-of-Tasks styles
    - exactly-once message delivery



Tools

- *mper* probing engine
  - based on Matthew Luckie’s *sca**mper***
  - send/receive individual IPv4 ICMP, UDP, TCP packets
    - no traceroute or other high-level measurement functions
  - new control socket interface providing measurement API
    - write measurement scripts in Ruby (e.g., MIDAR)
    - Alistair King ported scamper's traceroute code to mper in Ruby

Tools

```
require 'mperio'

class Prober
  def initialize
    @mperio = MperIO.new 8742  # mper listening port
    @mperio.delegate = self
    @mperio.ping_icmp 1, "192.172.226.123",
                      :ttl, 3, :cksum, 0x1234, :rr, true,
                      :tsps, ["192.172.226.1", "192.172.226.2"]
    @mperio.start
  end

  def mperio_on_data(result)
    if result.responded?
      printf "%d %d\n", result.rx_sec, result.reply_ipid
    end
    @mperio.stop
  end
end
```

Tools

- *Dolphin*
    - conducts parallel PTR DNS lookups of IPv4 and IPv6 addresses
        - millions of lookups per day from a single host
    - retries failed lookups once per day for up to 3 days
    - ensures targets only looked up once in any 7 days regardless of TTL
        - reduces load on authoritative DNS servers
    - built on libunbound (part of Unbound by NLnet Labs)
        - a validating, recursive, caching resolver in a library; IPv4/IPv6/DNSSEC
    - hackable: single Python source file (845 lines)
        - no installation or root privileges required

Tools

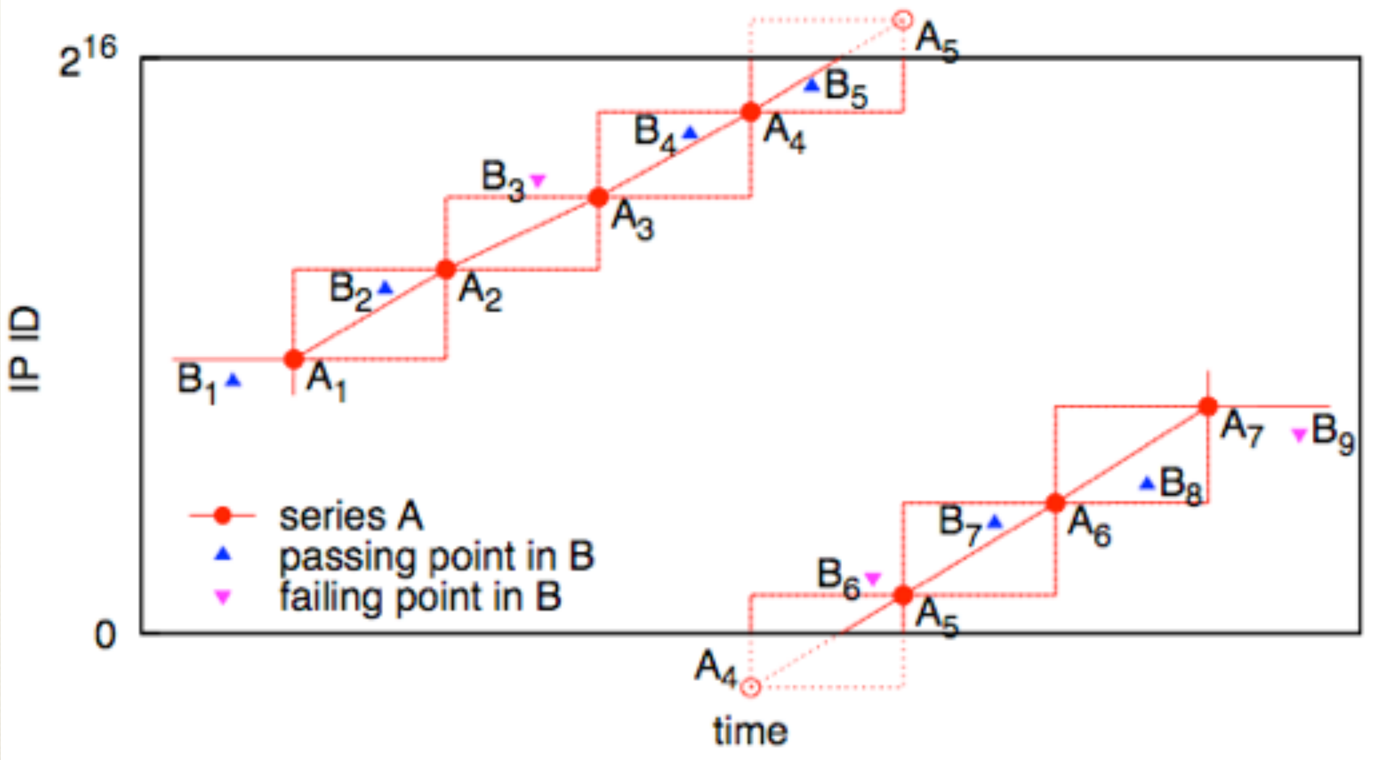
- *qr*
  - similar to Dolphin but more focused
    - only DNS lookups; no retries, no suppression of repeated lookups
  - supports PTR, SOA, A, AAAA lookups
  - uses *ldns* library for low-level structured access to raw DNS response packets
    - response header flags (e.g., AA)
    - records in authority and additional sections (e.g., glue, SOA, and DNSSEC records)
  - hackable: 513 lines of Python

Tools

- *qr* case study: PTR lookups of routed address space
  - 2.69 billion addresses (excluding .0 and .255 in each /24)
  - 3.6k queries/sec ⇒ 317M queries/day ⇒ 8.5 days
  - did full run in Aug 2014; data available

Tools

- *MIDAR*: Monotonic ID-Based Alias Resolution
  - **Monotonic Bounds Test**: for two addresses to be aliases, their combined IP-ID time series must be monotonic
  - 4 probing methods: TCP, UDP, ICMP, "indirect" (traceroute-like TTL expired)
  - sliding-window probe scheduling for scalability
  - multiple sources



Tools

- *tod-client*: on-demand topology measurements
  - scriptable command-line interface for performing IPv4 and IPv6 traceroutes and pings

```
$ tod-client -h

1 san-us ping 192.172.226.123

ping from 192.172.226.5 to 192.172.226.123
  1:  192.172.226.123    0.092 ms  64 TTL
  2:  192.172.226.123    0.112 ms  64 TTL
  ...

2 lax-us trace 192.172.226.123

traceroute from 137.164.30.25 to 192.172.226.123
  1.1:  137.164.30.1    0.183 ms
  2.1:  137.164.46.105    0.787 ms
  3.1:  137.164.46.54    2.623 ms
  ...
```

Tools

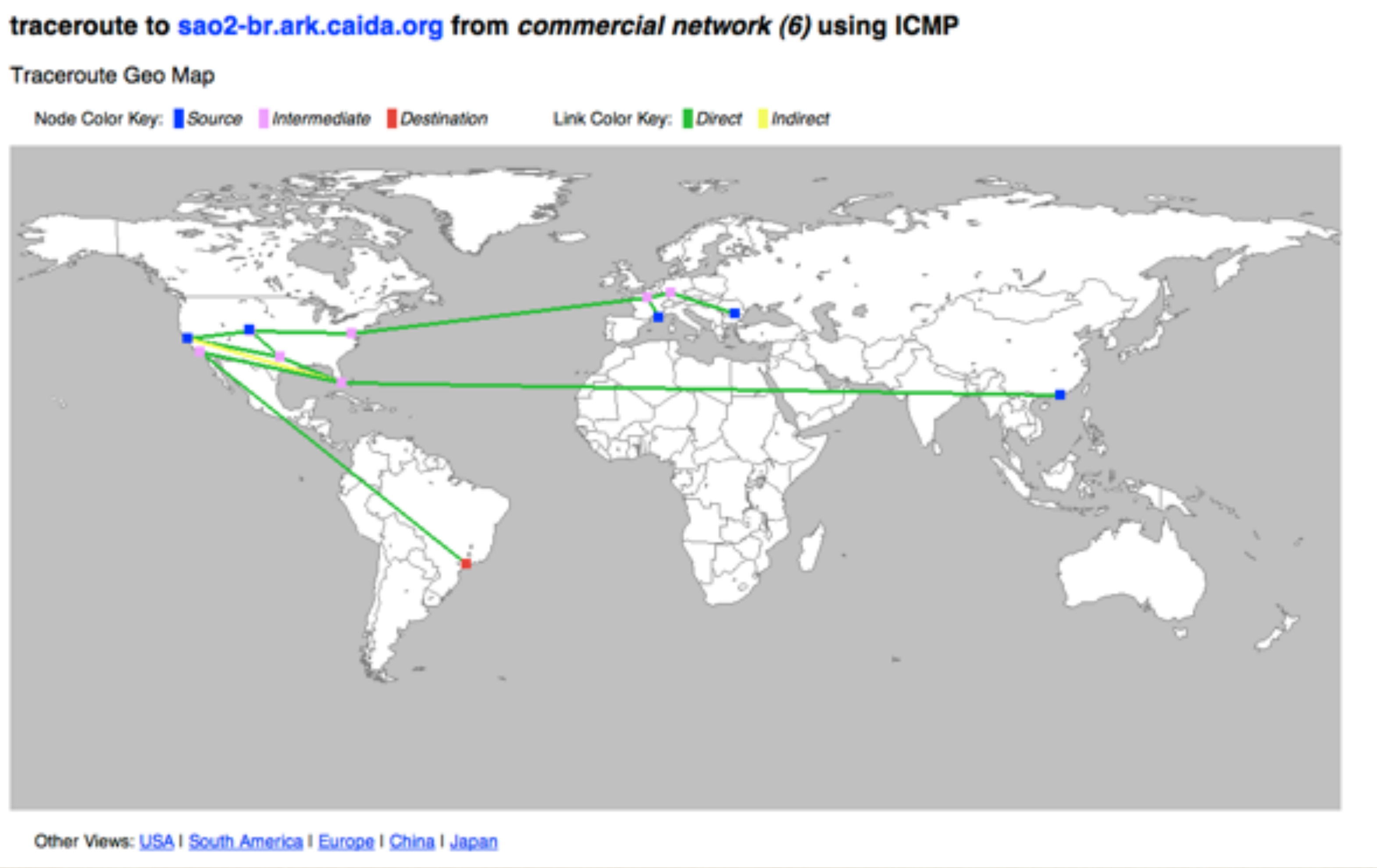
```
$ tod-client

1 san-us ping 2001:48d0:101:501::132 attempts=1

1 data 2001:48d0:101:501::132 P 2001:48d0:101:501::5
2001:48d0:101:501::132 0        1       1328149101      R
0.353   1       64      S       0
2001:48d0:101:501::132,0.353,64

2 lax-us trace www.caida.org attempts=1,method=icmp-paris

2 data www.caida.org T  137.164.30.25   192.172.226.123 0
1       1328145600      R       9.766   7       58      S
0       C       137.164.30.1,0.147,1
137.164.46.105,1.045,1  137.164.46.54,2.559,1
137.164.47.15,9.750,1  137.164.23.130,17.992,1
132.249.31.6,9.886,1
```

```
#!/usr/bin/env ruby

require 'marinda'

$tod = Marinda::Client.new "/tmp/localts.sock",
                           :port => 2000, :scope => :global

# 2 lax-us trace www.caida.org attempts=1,method=icmp-paris
$tod.write ["TRACEROUTE", "ark", 2, "lax-us", "www.caida.org",
            [["attempts", 1], ["method", "icmp-paris"]]]
result = $tod.take ["RESULT", "ark", nil, nil, nil, nil, nil]
p result
```

```
$ ./tod-example
["RESULT", "ark", 2, "lax-us", "www.caida.org", "data", "T
\t137.164.30.25\t192.172.226.123\t0\t1\t1328226507\tR\t9.838\t7
\t58\tS\t0\tC\t137.164.30.1,0.176,1\t137.164.46.105,1.110,1
\t137.164.46.54,3.015,1\t137.164.47.15,9.681,1
\t137.164.23.130,10.178,1\t132.249.31.6,9.860,1"]
```

Tools

- *Vela:* web interface to conduct topology measurements
  - currently, ping and traceroute (ICMP, TCP, UDP)



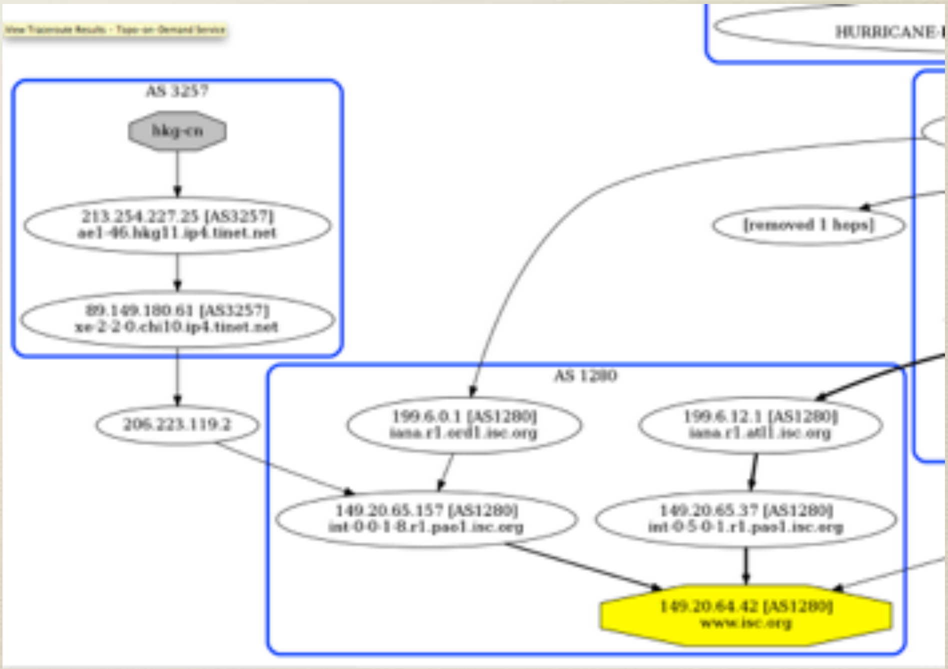
traceroute to sao2-br.ark.caida.org from *commercial network (6)* using ICMP

Traceroute Geo Map

Node Color Key: 🟦 *Source* ▨ *Intermediate* 🟥 *Destination*

Link Color Key: 🟩 *Direct* 🟨 *Indirect*

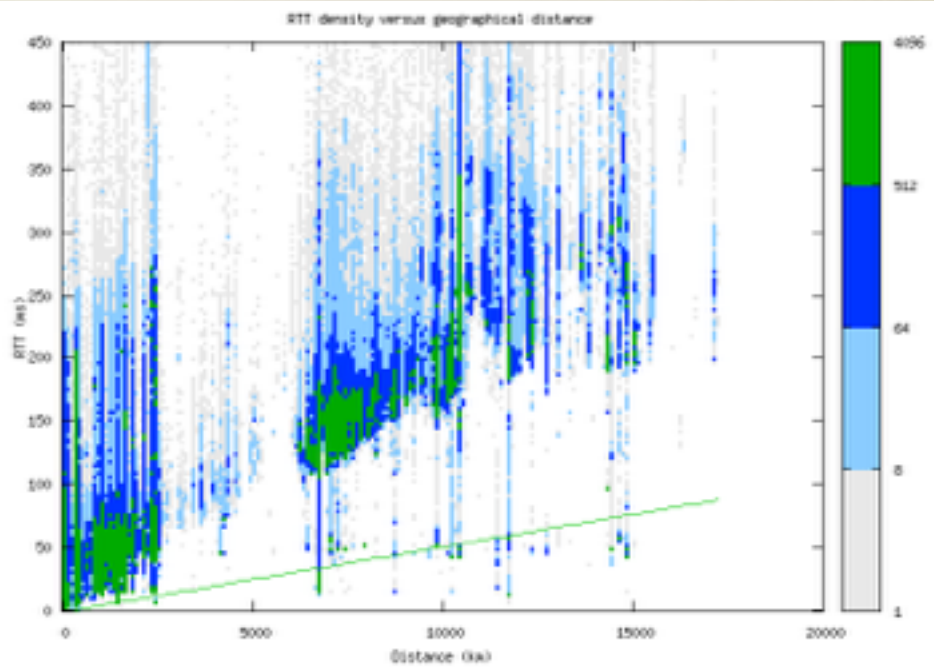
Other Views: USA | South America | Europe | China | Japan



View Traceroute Results - Topo-on-Demand Service

Measurements

- IPv4 topology
  - traceroutes to random address in each routed /24
  - 570 million traces/month
- IPv6 topology
  - traceroutes to random address and ::1 in each routed prefix
  - pings to IPv6 addresses of Alexa top 1 million sites
  - 16 million traces/month
- PTR DNS lookups of observed IPv4 and IPv6 addresses



Measurements

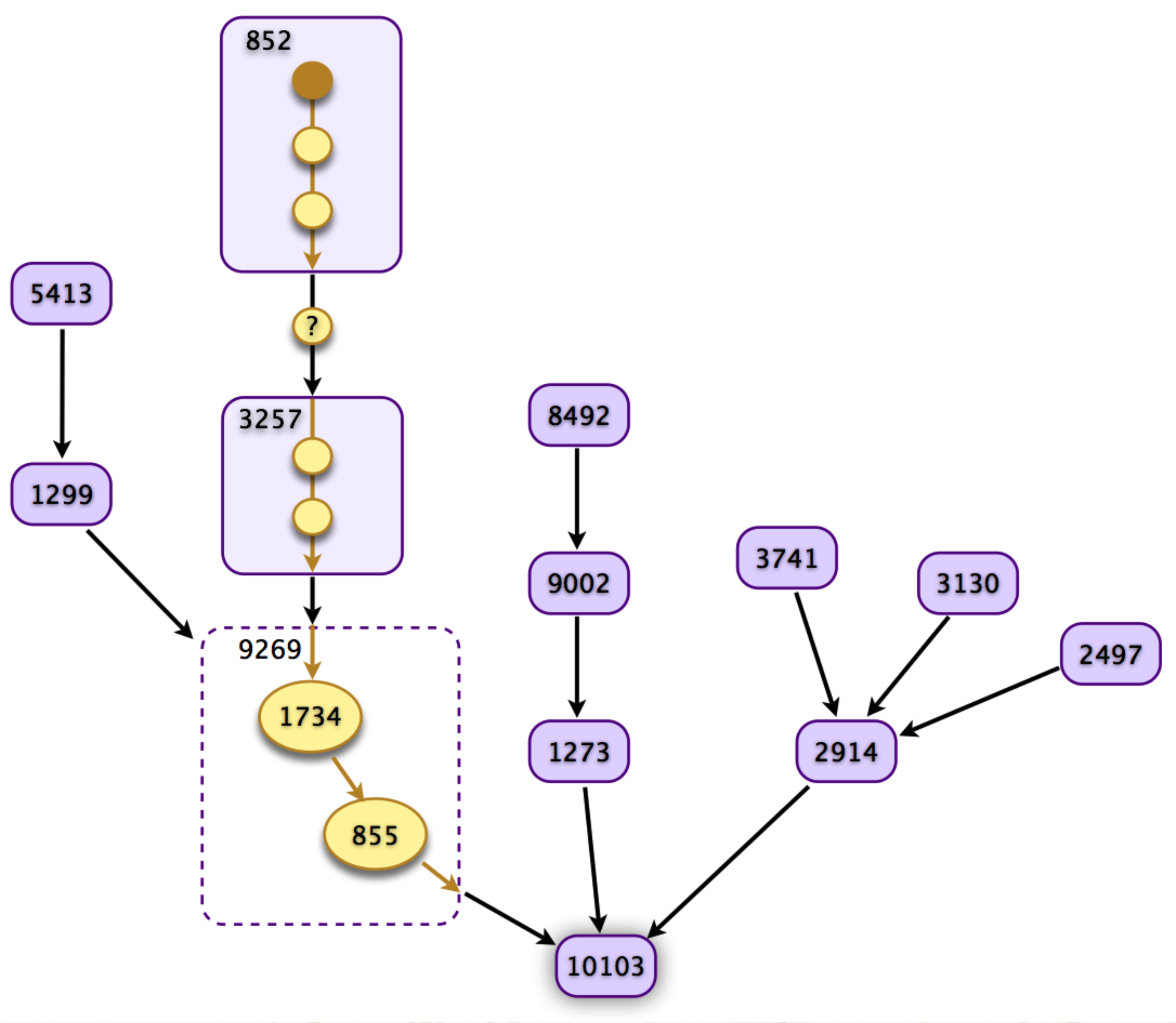
- alias resolution
  - MIDAR: collects IP-ID time series with TCP, UDP, and ICMP
  - iffinder: elicits ICMP port unreachable with UDP
- congestion at inter-domain peering links
  - elicits ICMP TTL-expired at adjacent IP hops
  - look for jumps in RTT across links

Ark Usage

- multiple ways of using Ark
  - **simplest**: Vela
  - **more control**: tod-client
    - example: Rob Beverly's IPv6 subnet topology discovery technique
  - **full control + high packet rates**: shell access
    - standard desktop/server Unix environment (not embedded)
    - raw socket access; no modifications required (no secure raw sockets layer)
    - compile and run any existing Unix program
    - write measurements in Ruby with Ark software
    - examples: middlebox study, Speedtrap IPv6 alias resolution, Casey Deccio's cctld DNS study (with dnsget)

Future

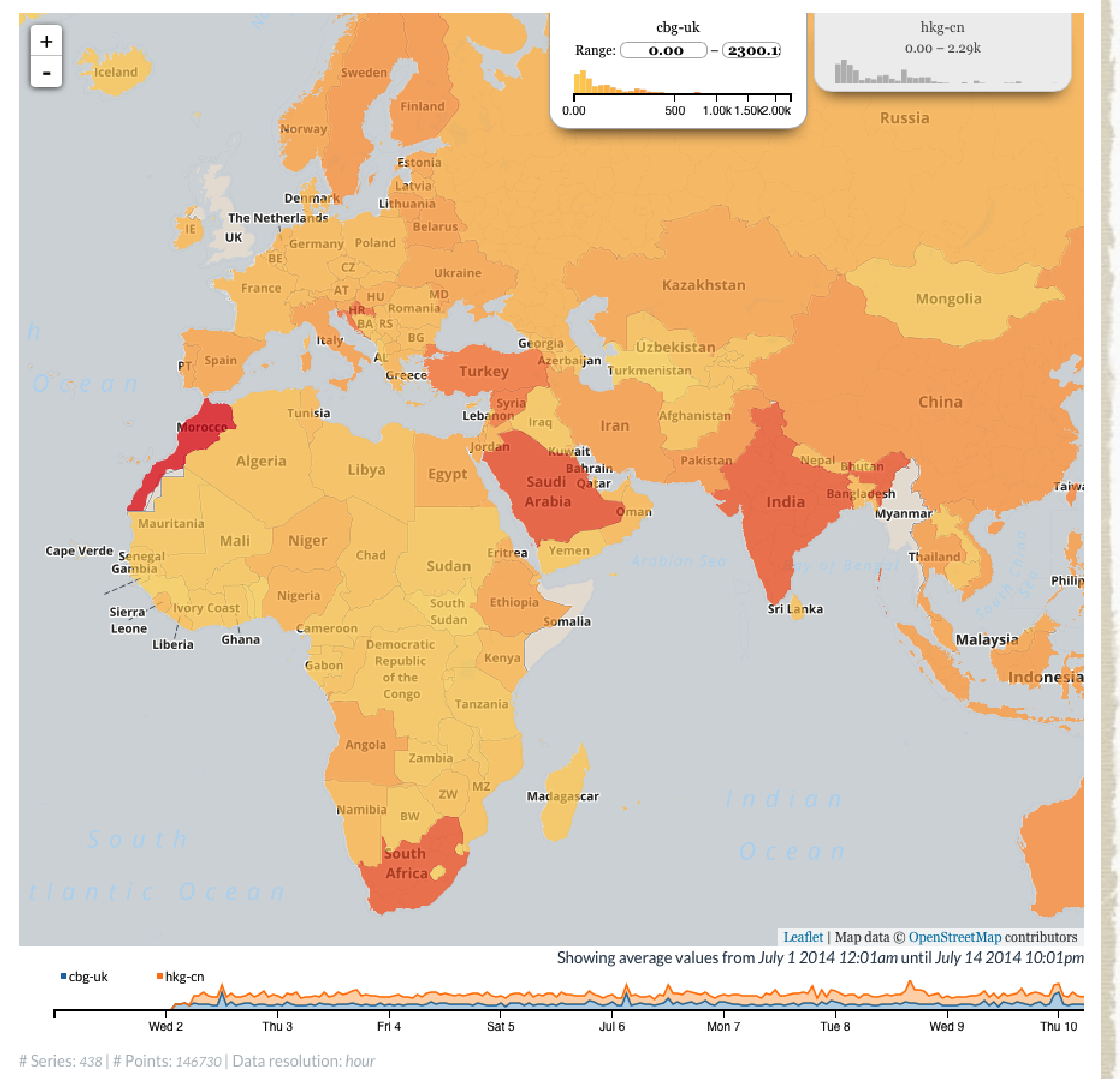
- improve data accessibility
  - create an interface for **browsing**, **querying**, and **visualizing** the data gathered by the infrastructure
  - command-line and web interfaces



prototype viz showing differences between a traceroute path and BGP AS paths

Future

- *browsing* interface
  - view broad properties and summary statistics over multiple time scales and aggregation levels
    - example: trace counts and response rates; path-length and RTT distributions; inferred AS links



Showing average values from *July 1 2014 12:01am* until *July 14 2014 10:01pm*

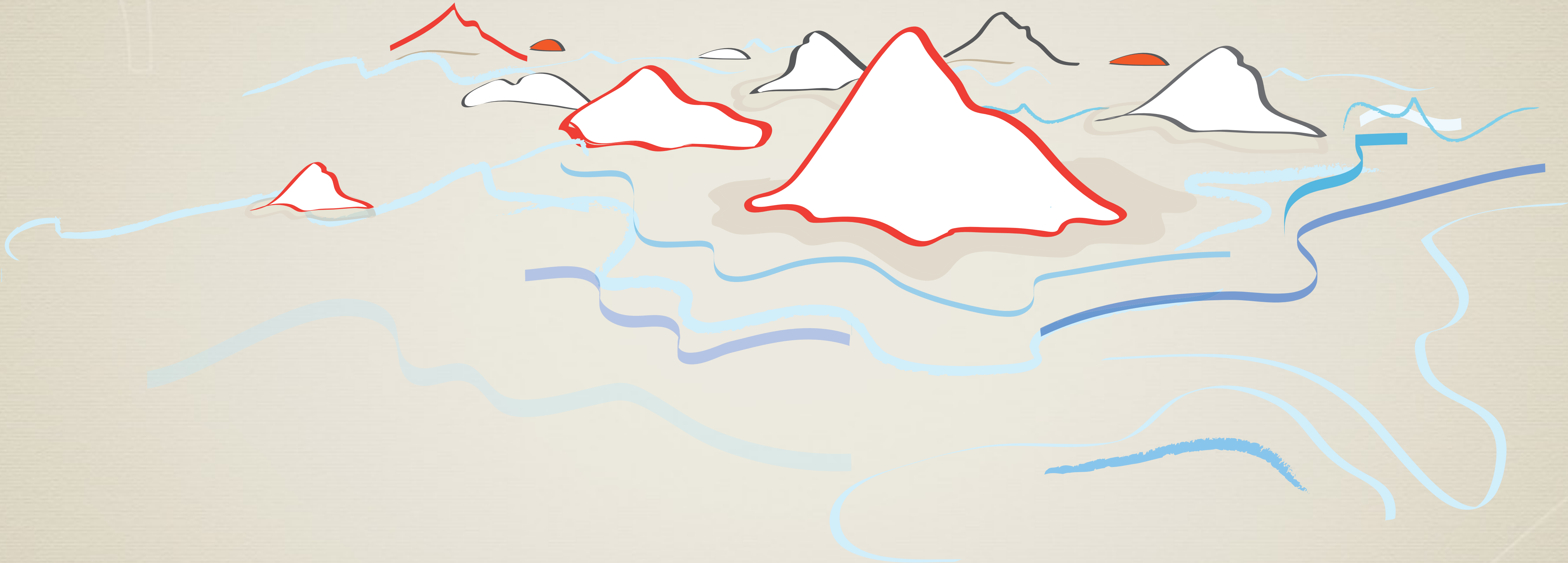
Series: *438* | # Points: *146730* | Data resolution: *hour*

prototype view of traceroute RTTs implemented with CAIDA's Charthouse

Future

- *query* interface
  - find the most relevant historical data for one's research
    - either directly answers a question, or identifies data to download for further study
  - examples:
    - all traceroutes through a given region and time period toward/across a particular prefix/AS
    - router address aliases for a given IP address
    - all inferred links to a router identified by a given IP address
    - all routers in a given city

Thanks!



www.caida.org/projects/ark

For questions, or to offer hosting: ark-info@caida.org