Crowdsourcing ISP characterization to the network edge

Fabián E. Bustamante

EECS, Northwestern U.

On the ground …

Mario Sanchez

David Choffnes (@ UWash)

Zach Bischof

John Otto

…

Need to get back to this

ISP characterization

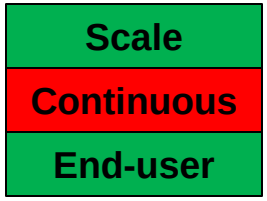
- To understand the configuration, policies and quality of service of access network service providers
- *Who needs it?*
  - Subscribers shopping for alternatives ISPs
  - Companies providing reliable Internet services
  - Governments surveying the availability of Internet to their citizens

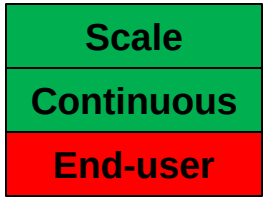
ISP characterization

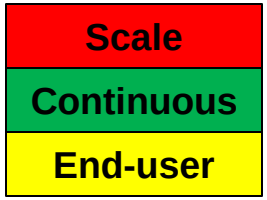
- *How should it be done?*
  - At scale – To capture diversity of providers and services
  - Continuously – To capture dynamics due to management policies, unscheduled events, evolution …
  - By end users – To guarantee its accuracy

Existing approaches to characterization

- Web-based technology test against dedicated or cloud servers
  - E.g. Netalyzr, Speedtest, YouTube/my_speed, …
- End-host monitoring from dedicated servers
  - E.g. Dischinger et al., Croce et al.
- Installing special monitoring devices at PoPs or home networks
  - E.g. SamKnows and FCC, Keynote



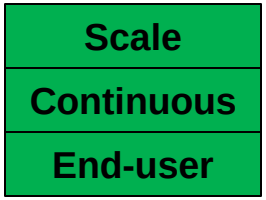




- *An unavoidable tradeoff between vantage points, coverage and continuous monitoring?*

Engaging the crowd at the network edge

- Leverage the views of Internet-wide ISP performance from popular networked apps
- Our current hosting application – *BitTorrent*
- Scalability and coverage from monitoring an application that growth with the network edge
- Continuously for an ISP
- Capturing the real performance end users receive



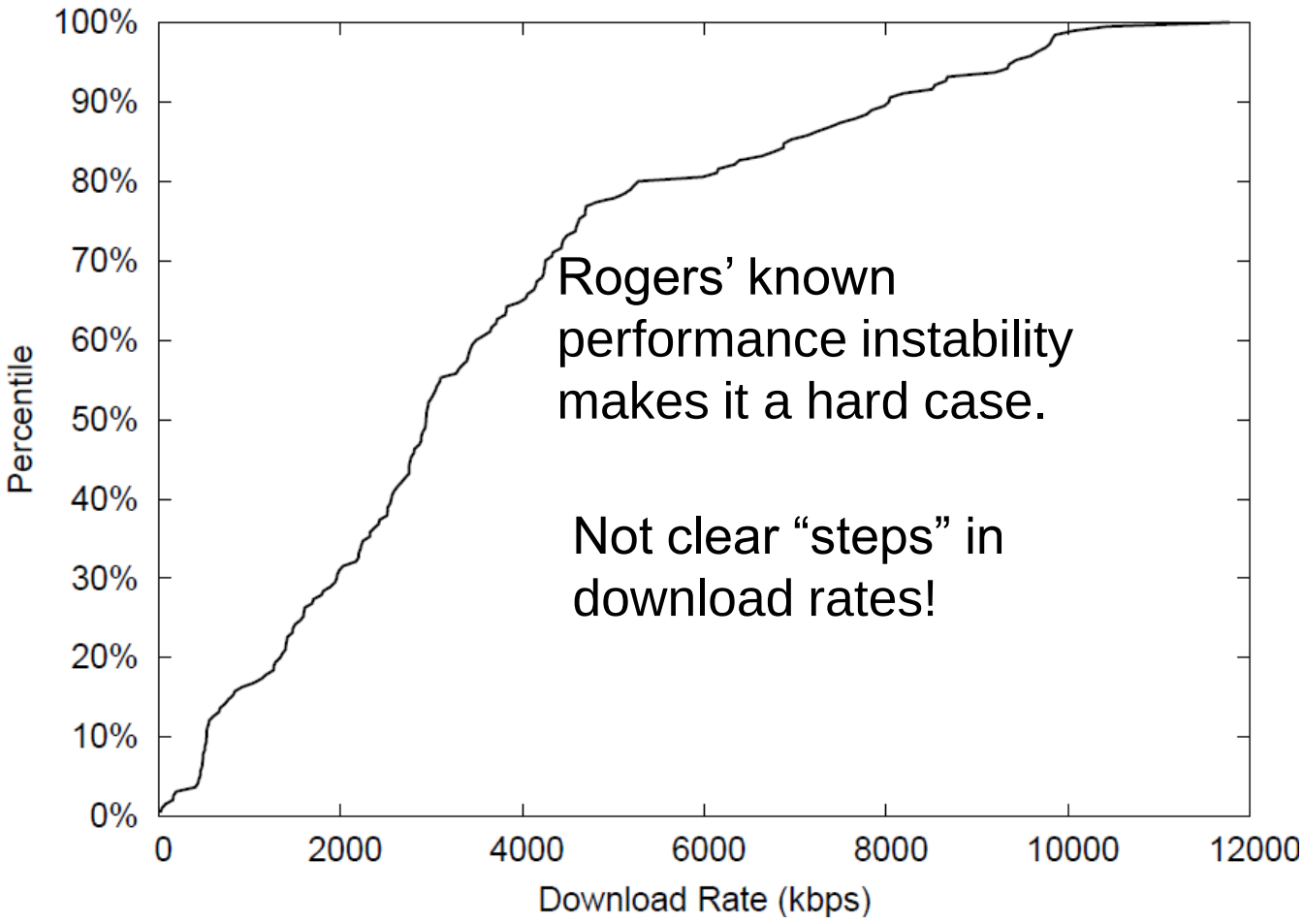
A quick roadmap

- Feasibility, of sorts
  - Can we do it from within an application?
  - Capturing performance dynamic variations
  - Capturing space variations
- Going beyond characterization
- Dasu - a new platform for ISP characterization from the edge

Can you do it from within BitTorrent?

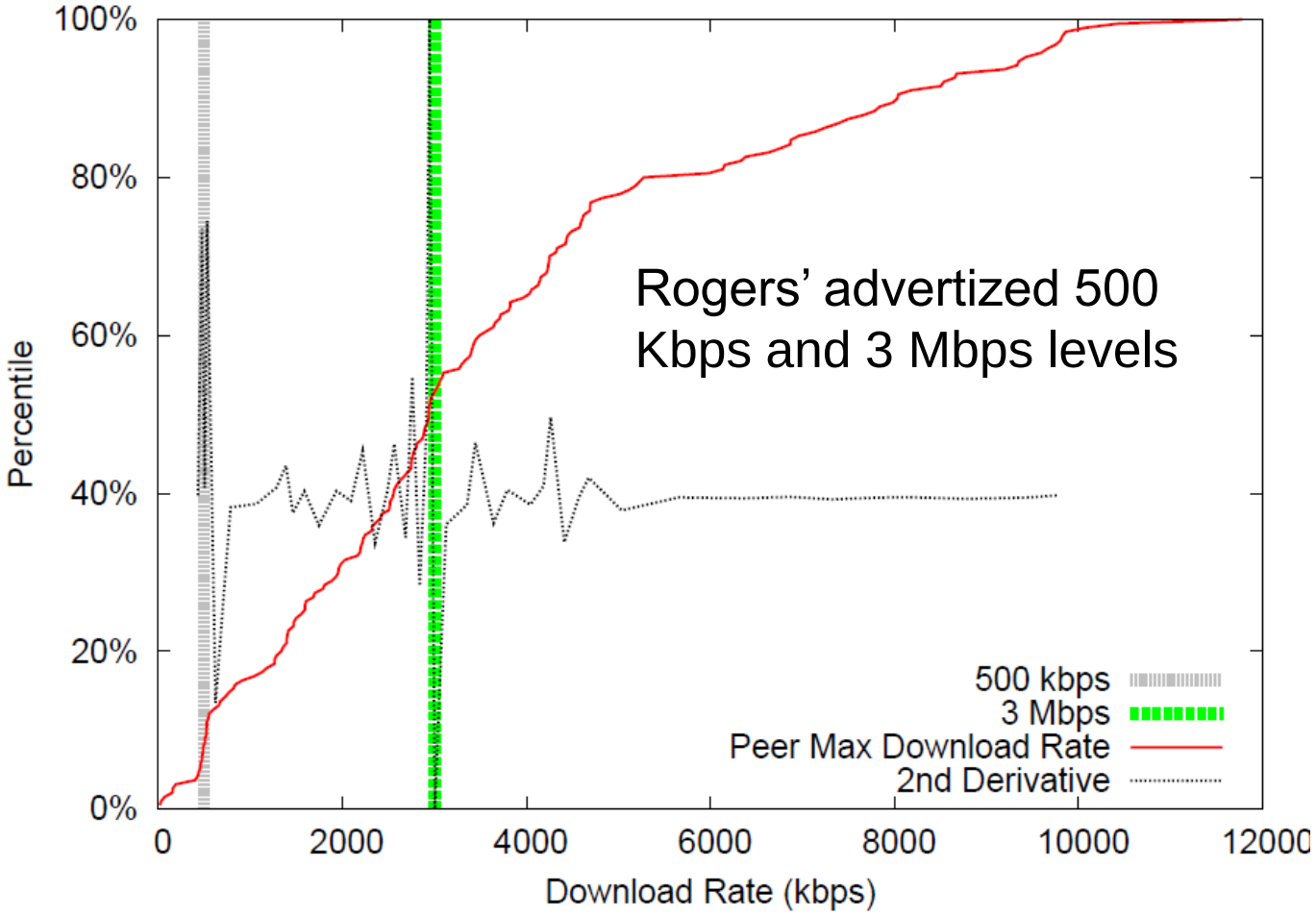
- Could application effects impede characterization?

Download rate of BitTorrent users in Rogers



ISP service levels

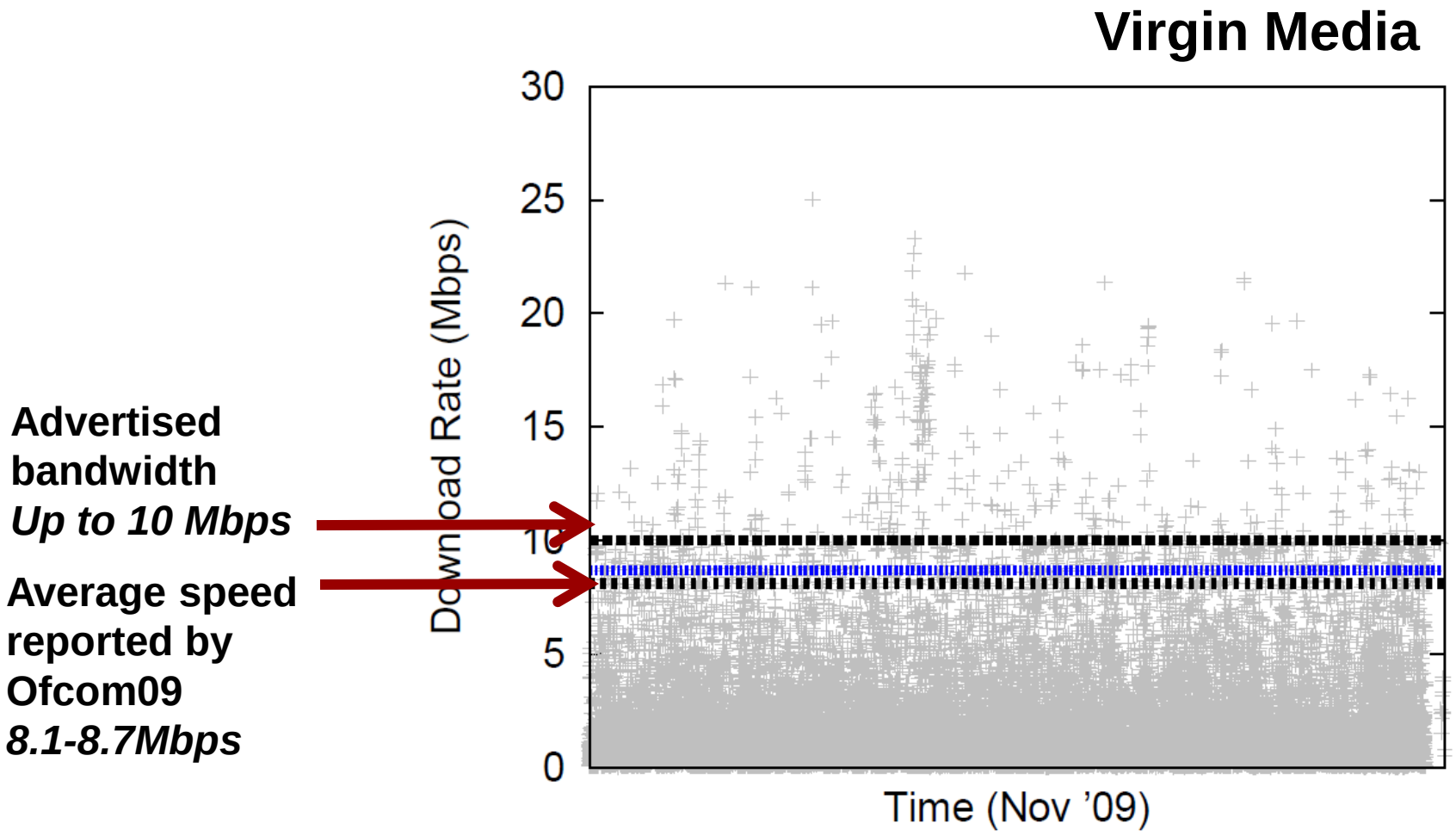
- Extracting Rogers' service levels



Scale
Continuous
End-user

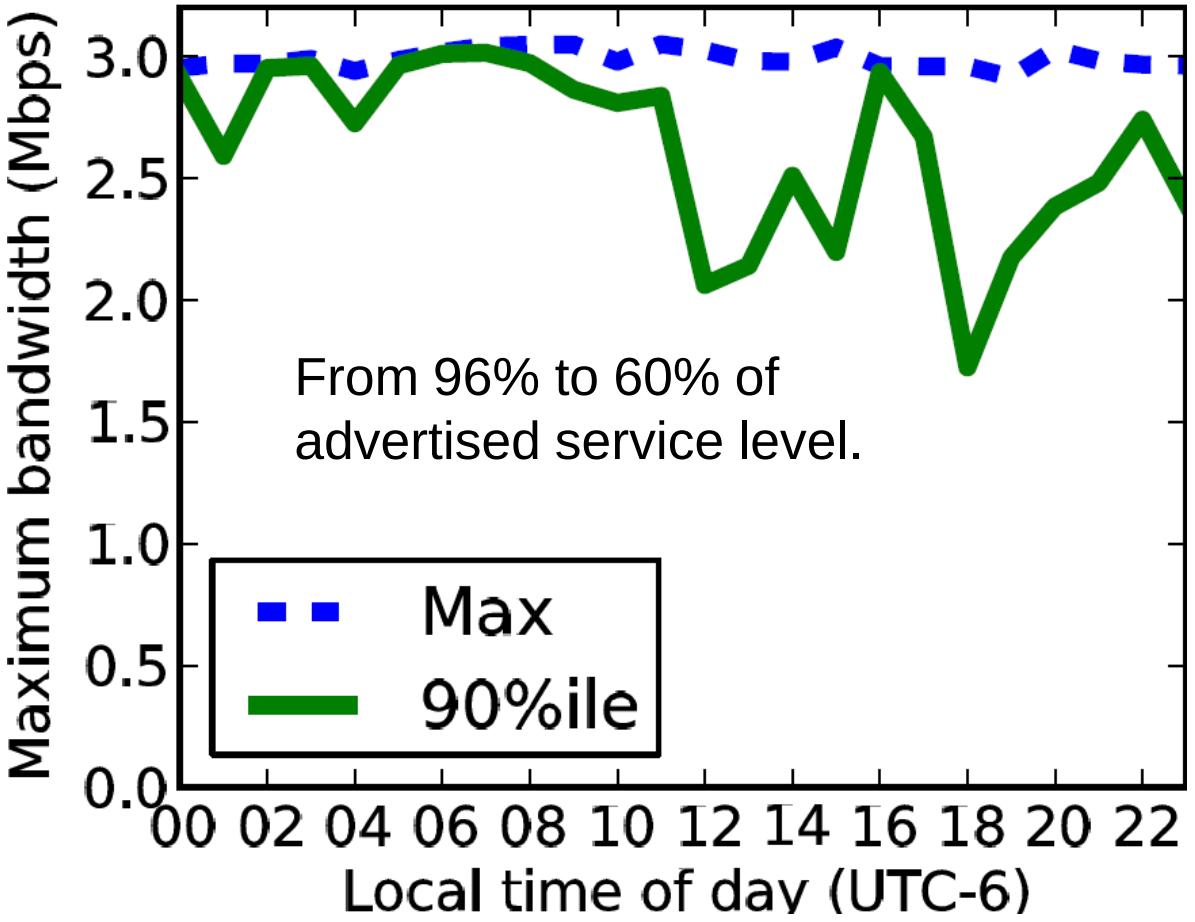
Comparing with a hardware-based approach

- Observed ISP performance and that captured by SamKnow's "white box"



Capturing service variations over time

- Variations on Rogers performance during the day (aggregated over Nov. 2009)

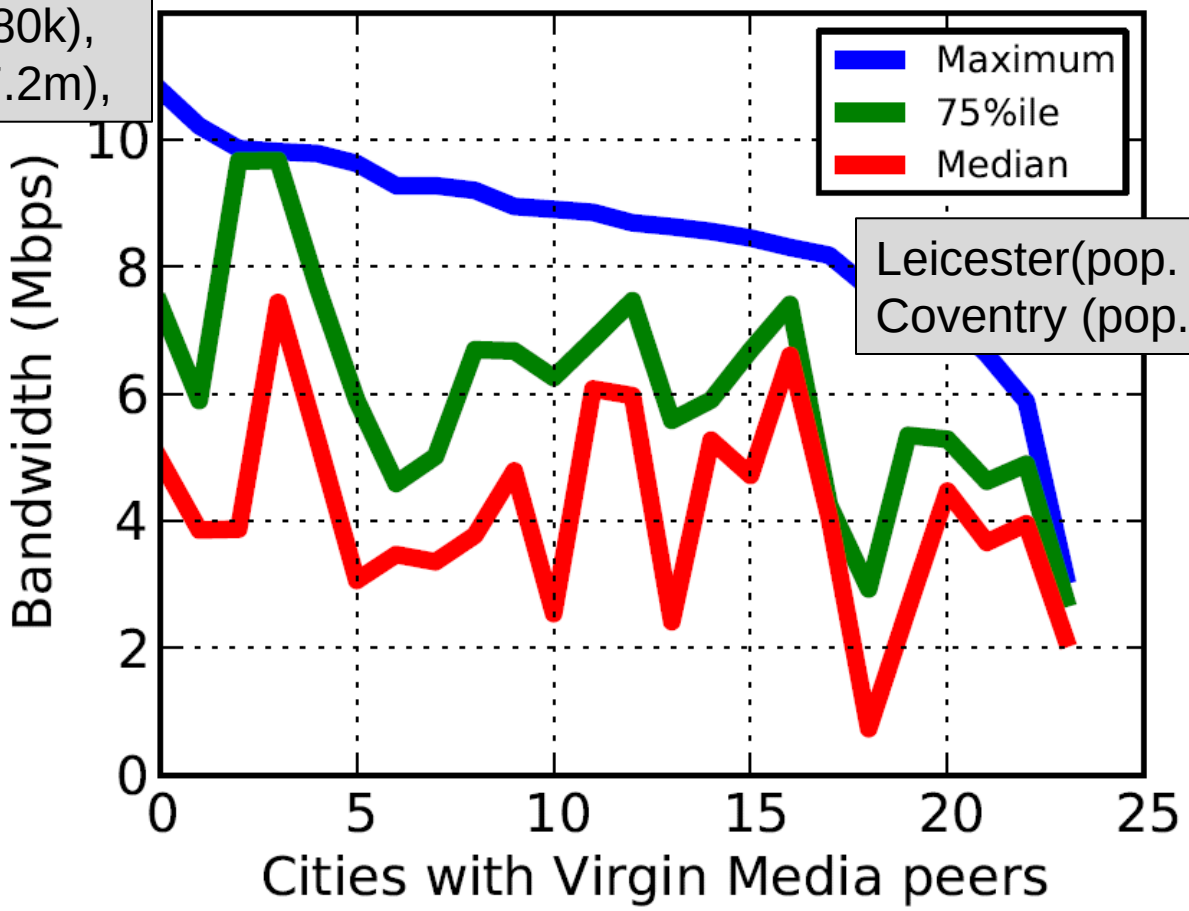


Scale
Continuous
End-user

Service variation across geography

- Variations on service levels among Virgin Media covered UK cities (order by maximum)

Belfast (pop. 280k),
London (pop. 7.2m),

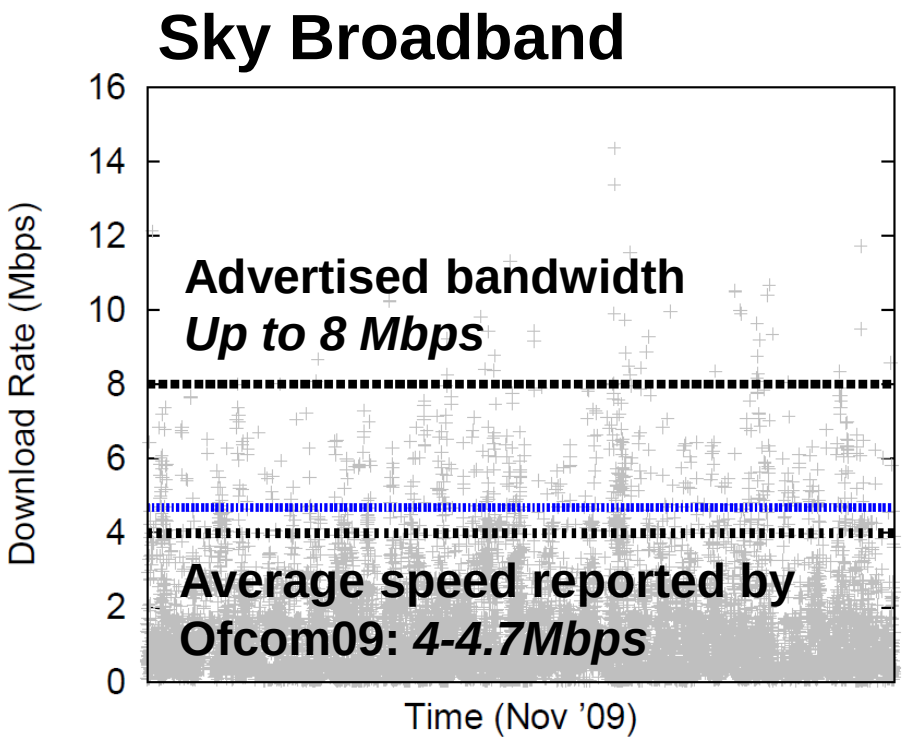


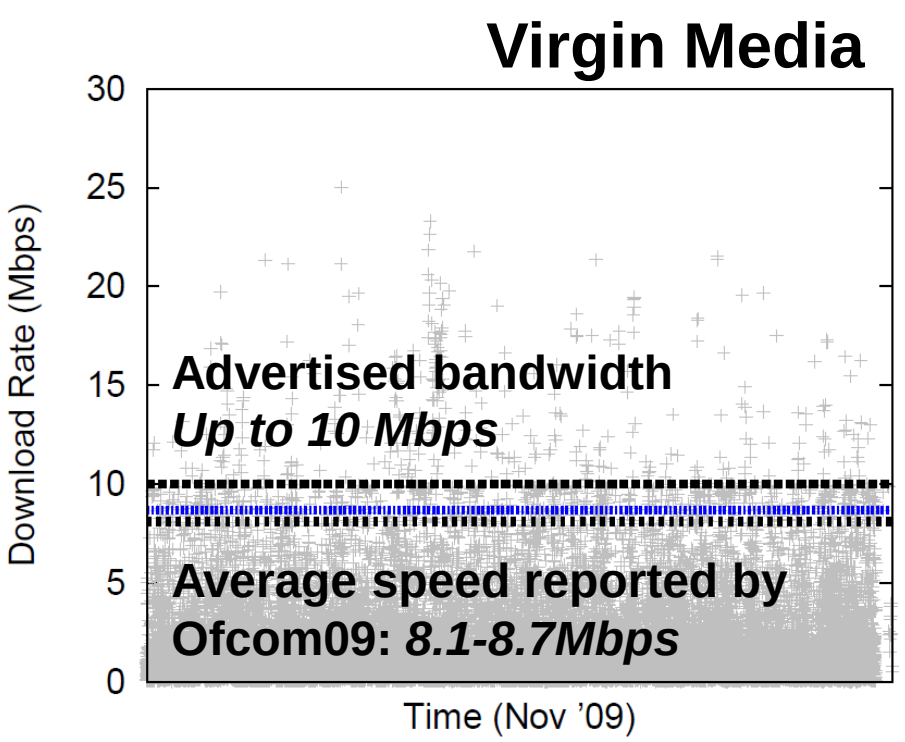
Leicester(pop. 280k),
Coventry (pop. 300k),

Scale
Continuous
End-user

Beyond characterization – Comparing ISPs

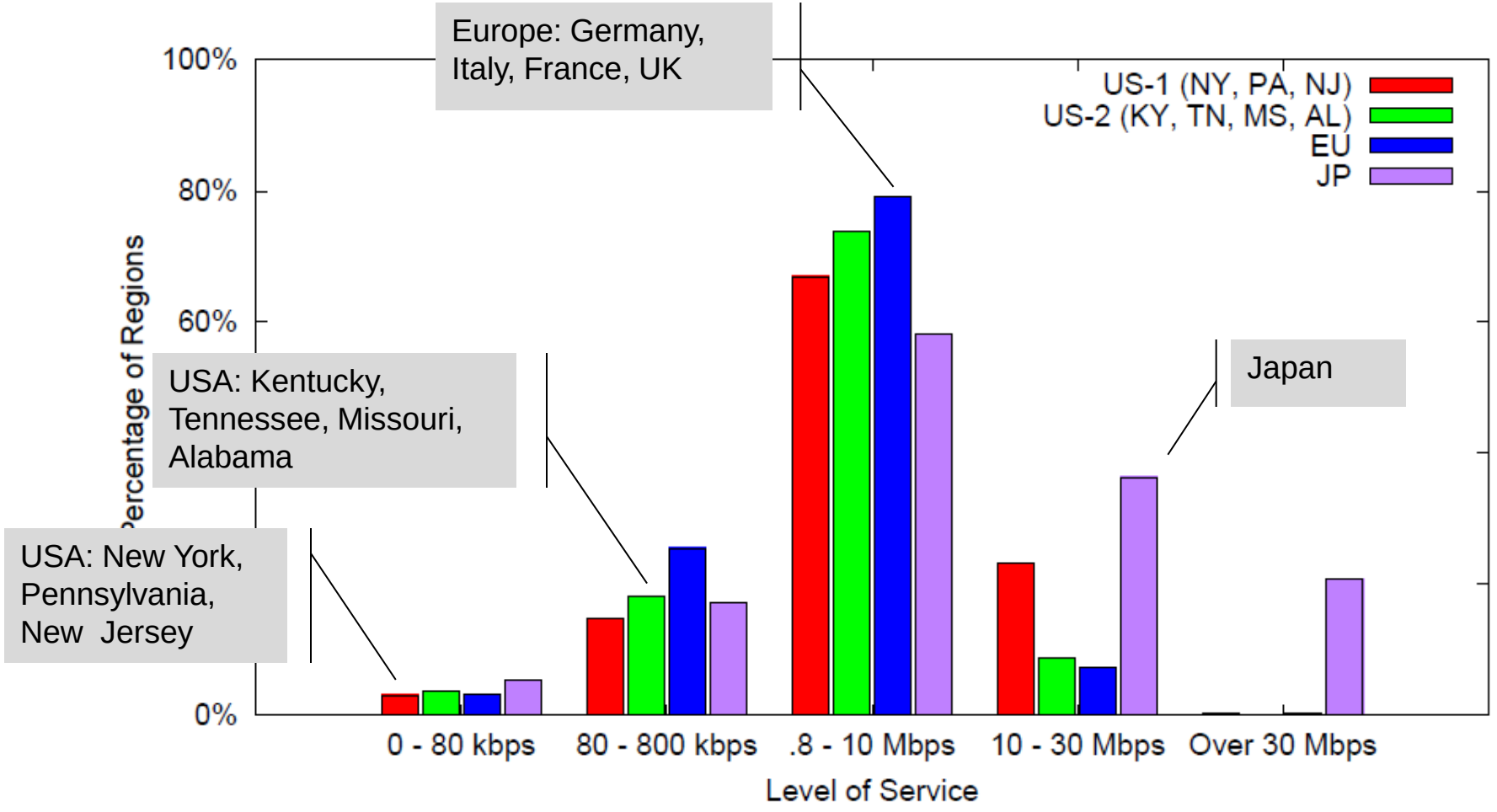
- Observed ISP performance and that captured by SamKnow's "white box"





Beyond characterization – Broadband studies

- Percentage of sub-regions containing at least one ISP providing each level of service



Dasu – A platform for ISP characterization

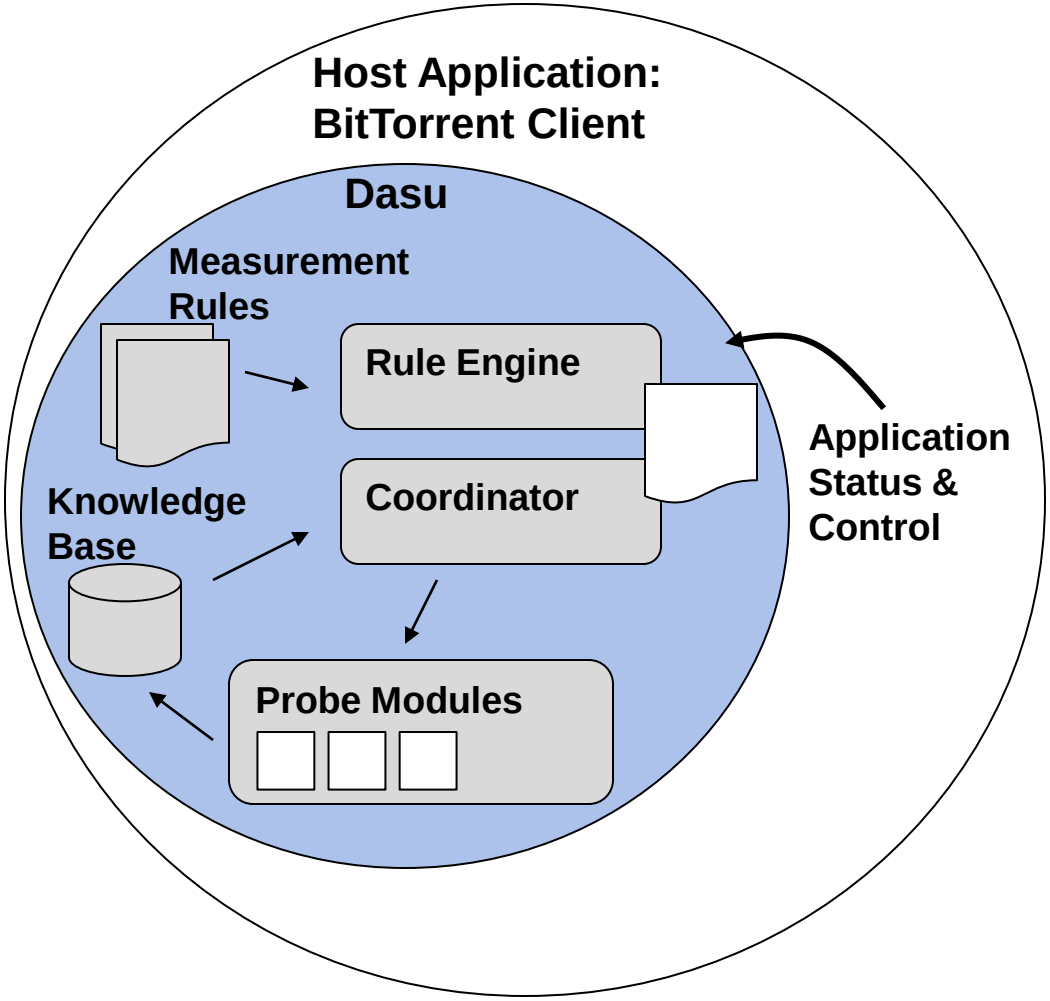
- A new extension to BitTorrent Vuze
- Combine passive and controlled active monitoring
  - Passive to capture end user's view in a scalable manner
  - Controlled active to avoid application-specific bias and for validation
- Enable dynamically extensible monitoring
  - To retain control, flexibility and low-barrier to adoption of software-based models
- Collaboration for eventual ISP comparison

Dasu prototype

Rule <name>
When {<condition>}
Then {<consequence>}

E.g.

```
rule "Launch BT test"
when
  $fact: something fishy found;
then
  addPriorityProbe("dload_n_encr",
                   ProbeType.BTTest);
  sendToLog("Launching BT Test");
  retract($fact);
end
```



Probe modules: traceroute, ping, ndt, dns, http get, …

Some details on monitoring rules

- General format

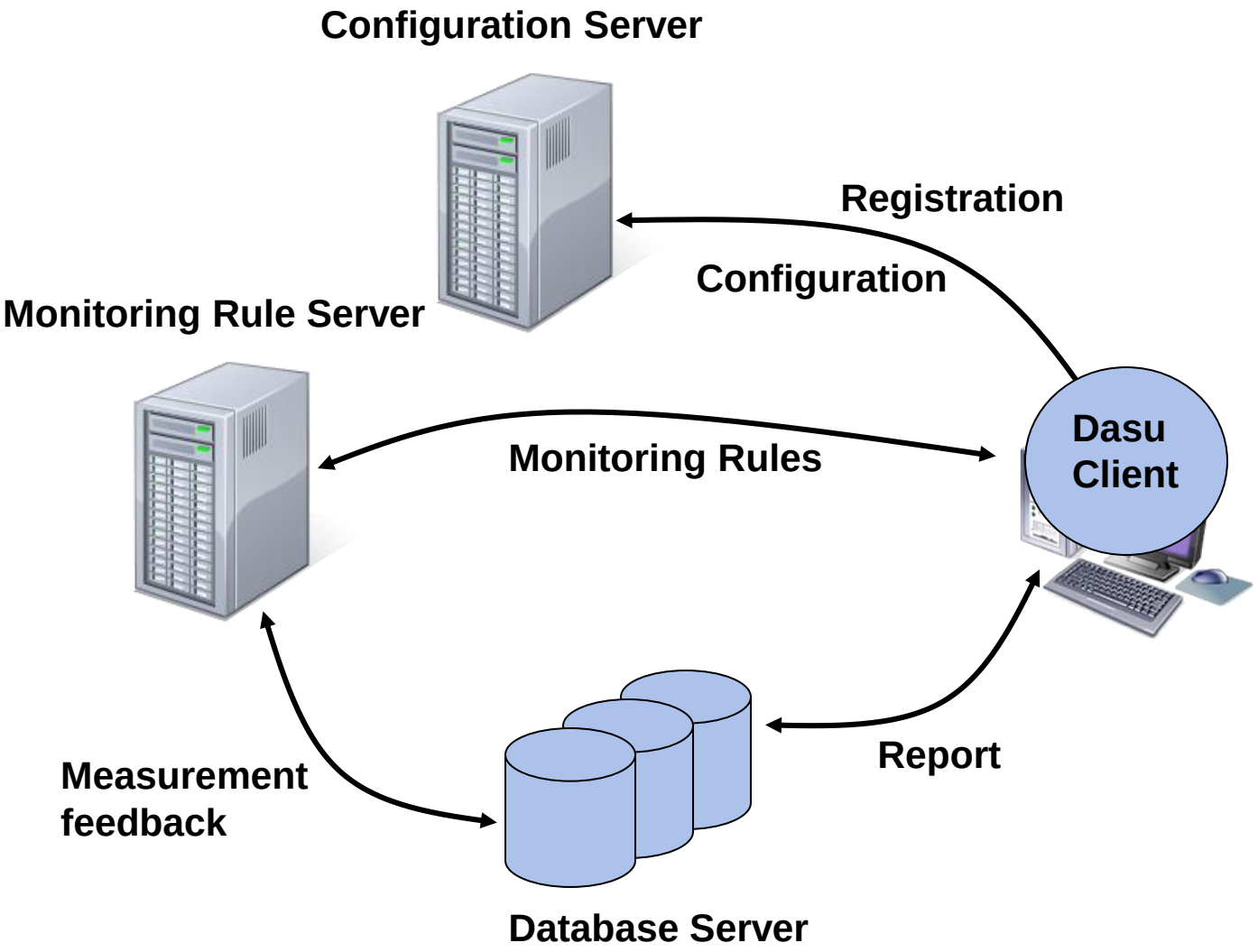
```
Rule <name>
When {<condition>}
Then {<consequence>}
```

E.g.

```
rule "Launch BT test"
when
  $fact: something fishy found;
then
  addPriorityProbe("dload_n_encr",
                   ProbeType.BTTest);
  sendToLog("Launching BT Test");
  retract($fact);
end
```

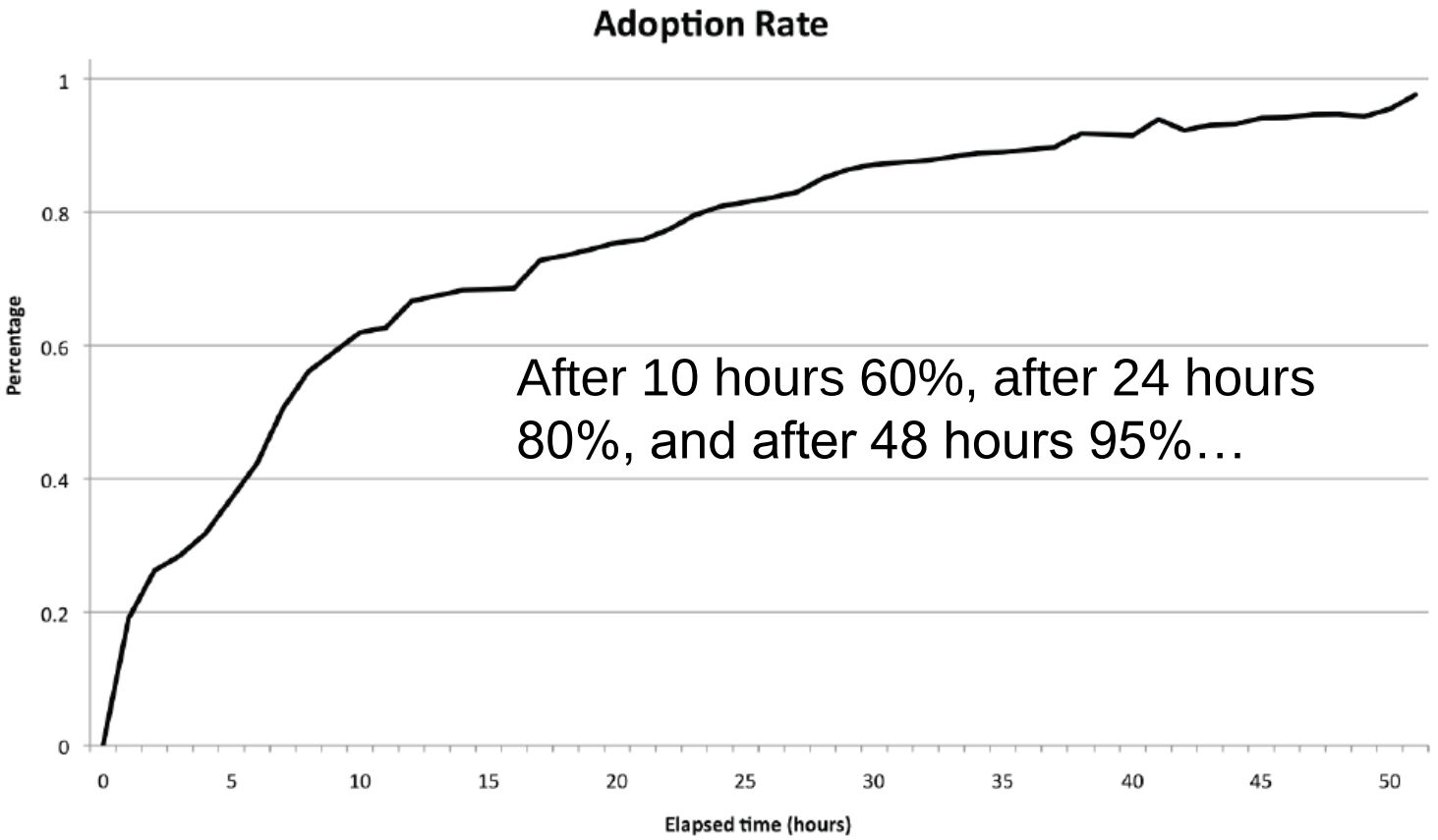
- Types of conditions
  - Facts in the knowledge base derived from passive, active monitoring and cron tasks
- Types of consequences:
  - Update knowledge base, launch new measurement, schedule new task, contact servers, plot results, …

Dasu prototype



Responsiveness to control

- Rules files are fetched when BitTorrent runs
  - So adoption rate determined by user inter-session times



Status

- First version released in June, 2010
- Without advertisement - > 25,000 users
- >1,000 ASes (>5,000 prefixes), 71% are eyeballs (growing at 25-43%)

Region	Growth	Dasu Growth	Dasu Countries
North America	146.3%	61%	3/5
Oceania/Australia	179%	58%	2/26
Europe	352%	60%	36/51
L. America/Caribean	1,032.8%	46%	16/24
Middle East	1,825.3%	47%	11/15
Asia	621.8%	48%	21/39
Africa	2,357.3%	55%	17/56

Summary

- ISP characterization needs to be done by end users, at scale and continuously
- Network intensive applications may provide a nearly ideal vantage point platform

- *What can we capture? What metrics should we use? Can we detect application biases? Can we compare ISPs? Can we handle "tricksy" ISPs? …*
- Exploring these and other questions with *Dasu*