RTG

A Scalable SNMP Statistics Architecture

USENIX LISA 2002

Robert Beverly

November 7, 2002

Overview

- Unique problems service providers & large enterprises face gathering statistics
- Discuss existing tools and limitations
- Introduce RTG
  - Architecture/features
  - Sample reports/output
- Questions

Background - What's the Problem

- SNMP: Simple Network Mgmt Protocol
- Despite "Simple," Many Issues:
  - Scaling in Large Installations
  - Storage Retention (Length/Granularity/Averaging)
  - Report Generation Time (Interactivity)
  - Reporting Flexibility
  - Robustness, statistics as a critical component:
    - Legal (Culpability)
    - Billing

Motivation

- Large Commercial Service Provider with 100’s of devices, 100’s of interfaces
- Other Open-Source packages could not complete polling within 5-minute interval
- New requirements to monitor additional per-interface statistics
- New reporting requirements

Requirements

- Four High-Level Requirements:
  - Support for 100’s of devices with 1000’s of objects (very high speed)
  - Ability to retain data indefinitely
  - Provide an abstract interface to data in order to generate complex and/or custom reports
  - Disjoined polling and reporting

Solutions

- Fix Existing Systems:
  - No clean separation of polling, reporting makes distributing load difficult
  - Faster hardware
- Commercial Package:
  - Large, bloated, expensive
- MRTG:
  - Scaling Problems
- Cricket + rrdtool:
  - Good scaling (can we do better?), no abstract data interface
- See Paper for Full Comparison/Analysis

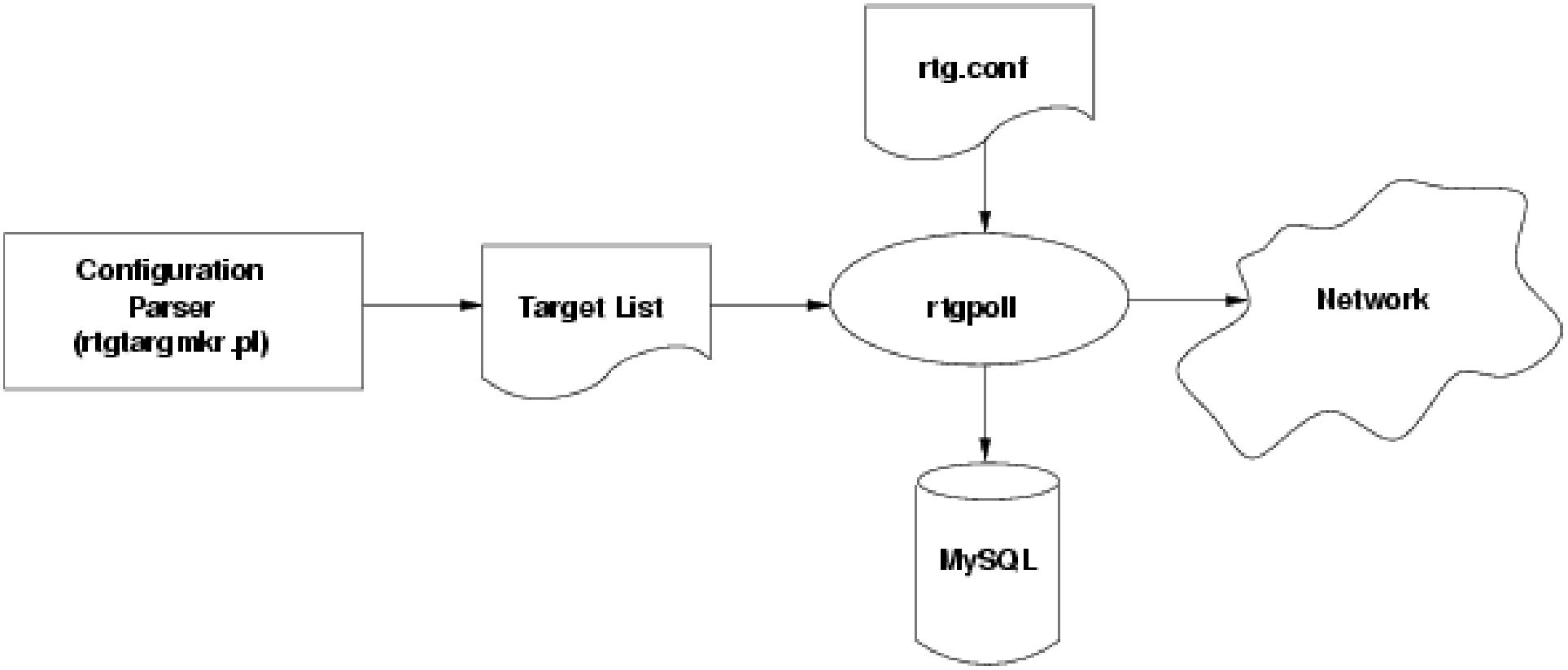
RTG: Real Traffic Grabber

- Flexible, scalable high-performance SNMP monitoring system
- Runs as a daemon on UNIX platforms
- Can poll at sub-one-minute intervals
- All data inserted into a relational database
- Keeps absolute samples, no averaging
- Intelligent database schema to retain long-term data without speed degradation
- Traffic reports, plots, web-interface
- Easily supports distributed polling, data redundancy

RTG Operation

- All data is inserted into the MySQL database
- Network configuration stored in database
- Auxiliary Perl script, “rtgtargmkr.pl” queries network for new interfaces and changed ifIndex or description.
- Generates an RTG “target list”
- RTG poller, “rtgpoll” randomizes objects in the target list
  - Limits SNMP query impact on network devices
  - Improves performance
- Reports and Graphs generated via APIs to MySQL (Perl DBI, PHP, C)

RTG Functional Diagram



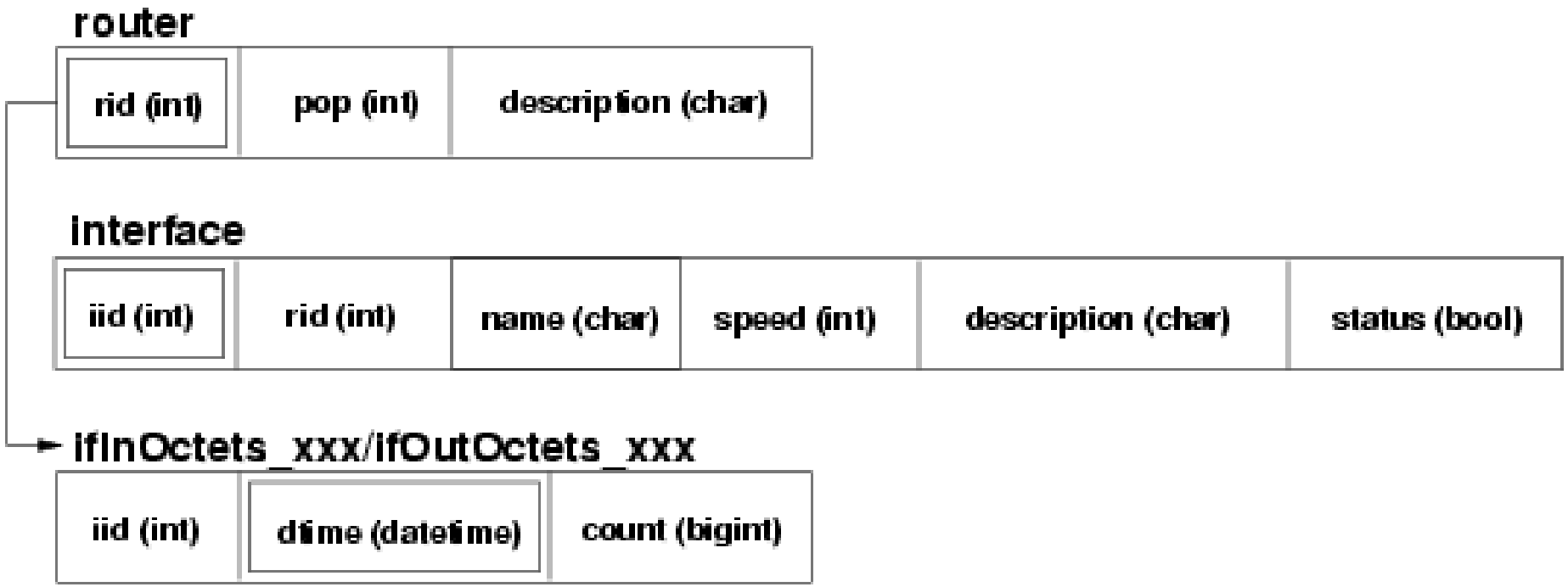
Database Schema

- Non-trivial
  - Better schemas for different environments
  - RTG poller is indifferent to schema
- Need to retain long-term historical data (ideally indefinitely):
  - Legal/Billing
  - Disks are cheap, but keep as little data as possibile
- Query execution time should be independent of time period requested:
  - Generating a report for a day one year ago should be as fast as generating today's report

Database Schema

- Router and Interface tables keep identifiers, descriptions, speeds, etc.
- Segment data as much as possible (indexes are great, but require index space)
- SQL table per unique device and object
  - ifInOctets_9 table
  - Store only date/time, sample and interface
- Index each table on date/time

Database Schema



RTG Speed

- What makes RTG fast?
  - Daemon – No cron overhead
  - Written in C – No interpreter overhead
  - Multi-threaded:
    - Keep a constant number of “queries-in-flight”
    - Exploit Natural Parallelism in Slow I/O
    - Use multiple processors
  - Randomized targets:
    - An unresponsive device does not block all threads

RTG Speed (Some Numbers)

App	Targets	Run Time (seconds)	Targs/sec	Max Targs
MRTG	1618	365.4	4.43	1328
Cricket	2010	87.8	22.89	6868
RTG	3650	34.2	106.73	32018

- Max Targets indicates theoretical maximum number of targets polled in a 5 minute interval

RTG Reports

- Perl DBI scripts included
- Automate reporting, etc.

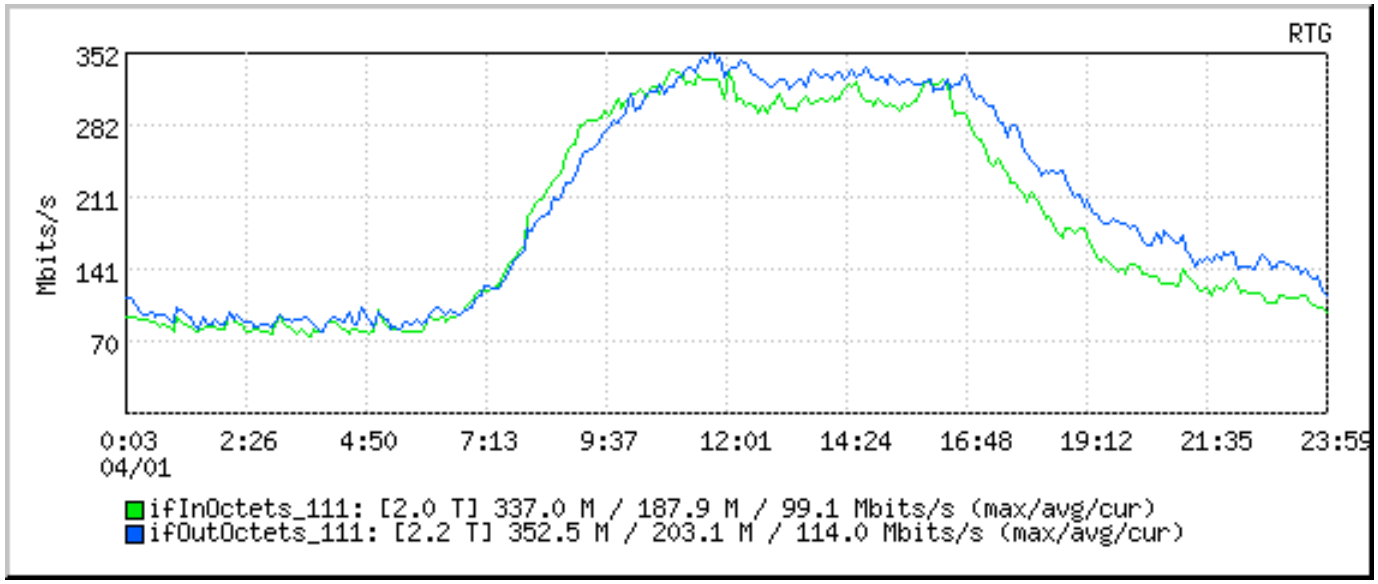
```
Traffic Daily Summary
Period: [01/01/1979 00:00 to 01/01/1979 23:59]

Site          GBytes In  GBytes Out  MaxIn(Mbps)   MaxOut    AvgIn    AvgOut
-------------------------------------------------------------------------------
rtr1.someplace:
so-5/0/0        384.734     360.857      49.013    43.420   35.630    33.426
so-6/0/0        357.781     421.736      42.923    50.861   33.137    39.053
t1-1/0/0          0.054       0.058       0.005     0.006    0.005     0.005
rtr3.someplace:
so-6/0/0      1,115.258   1,246.163     168.776   172.690  103.173   115.439
so-3/0/0      1,142.903   1,028.256     152.232   162.402  105.863    95.142
so-7/0/0        152.824     199.742      22.052    35.005   14.152    18.488
```

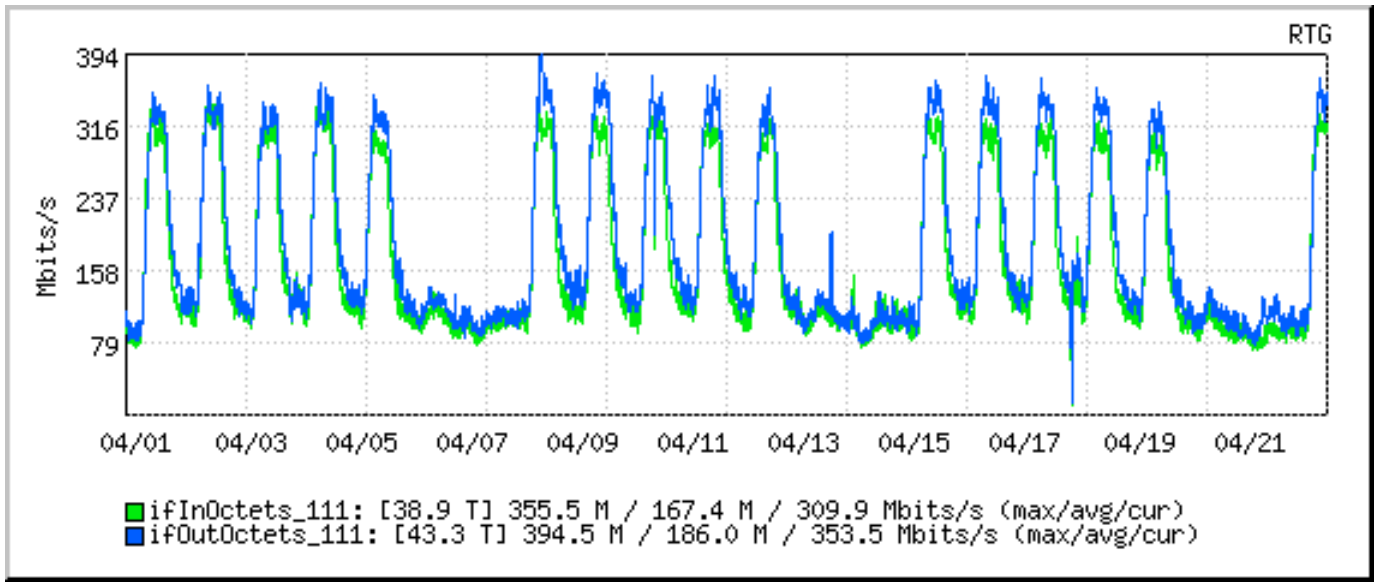
RTG Reports (95th Percentile)

ABC Industries Traffic
Period: [01/01/1979 00:00 to 01/31/1979 23:59]

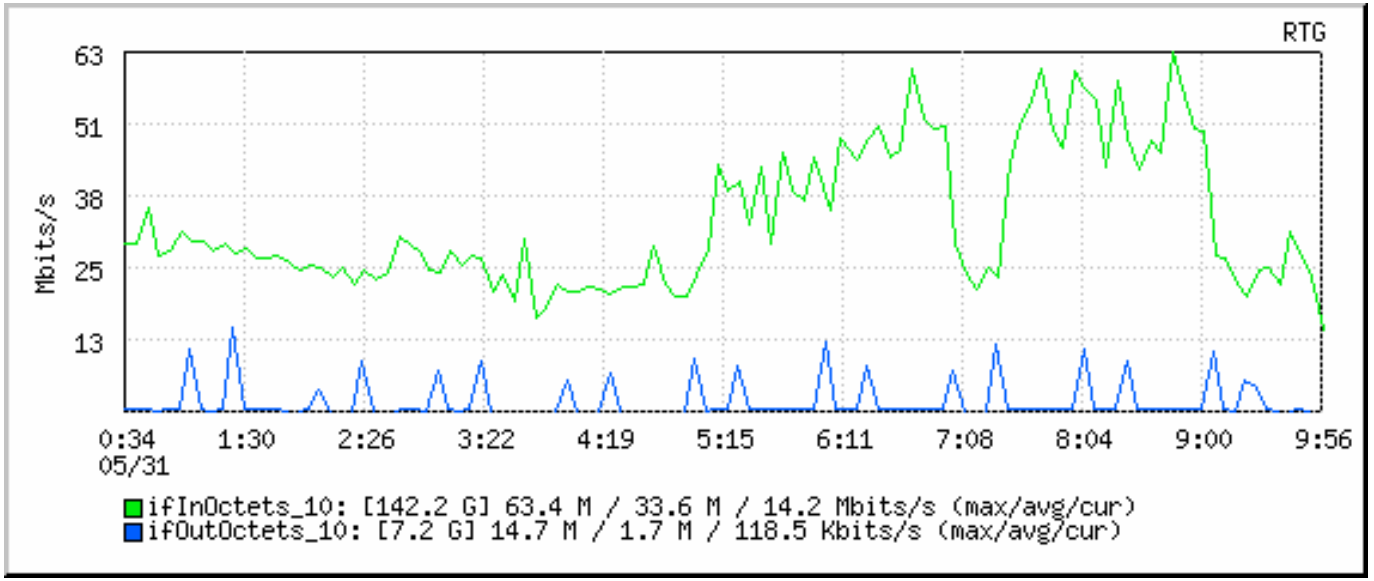
Connection	RateIn Mbps	RateOut Mbps	MaxIn Mbps	MaxOut Mbps	95% In Mbps	95% Out Mbps
at-1/2/0.111 rtr-1.chi	0.09	0.07	0.65	0.22	0.22	0.13
at-1/2/0.113 rtr-1.dca	0.23	0.19	1.66	1.12	0.89	0.57
at-3/2/0.110 rtr-2.bos	0.11	0.16	0.34	0.56	0.26	0.40

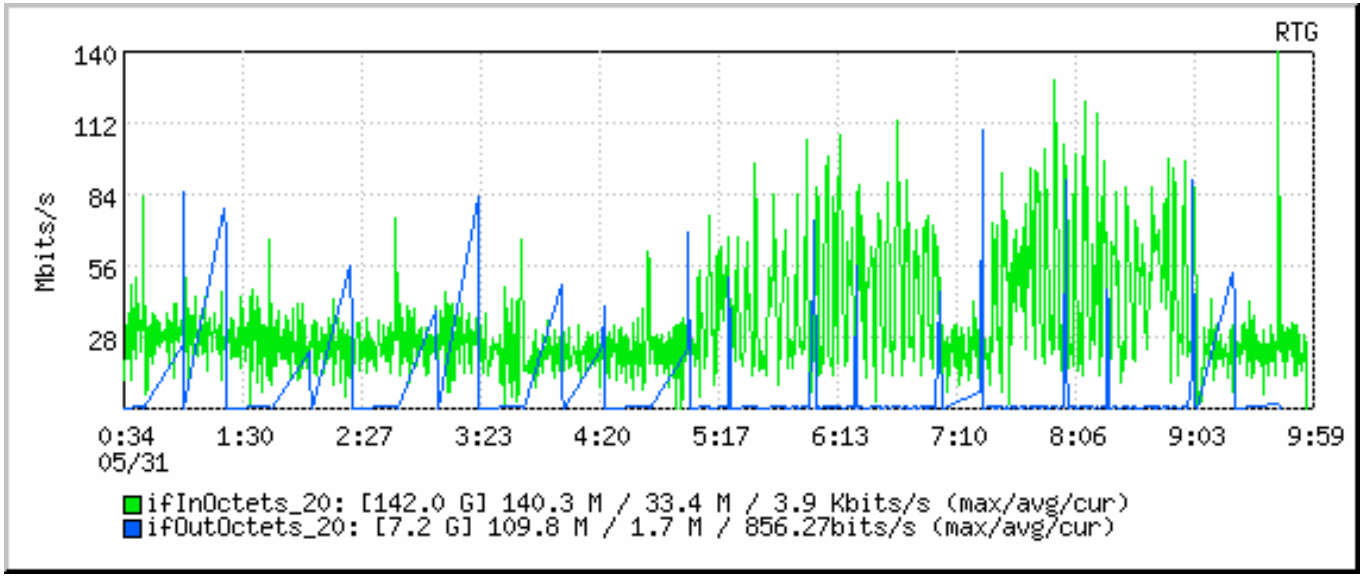
RTG Traffic Graphs





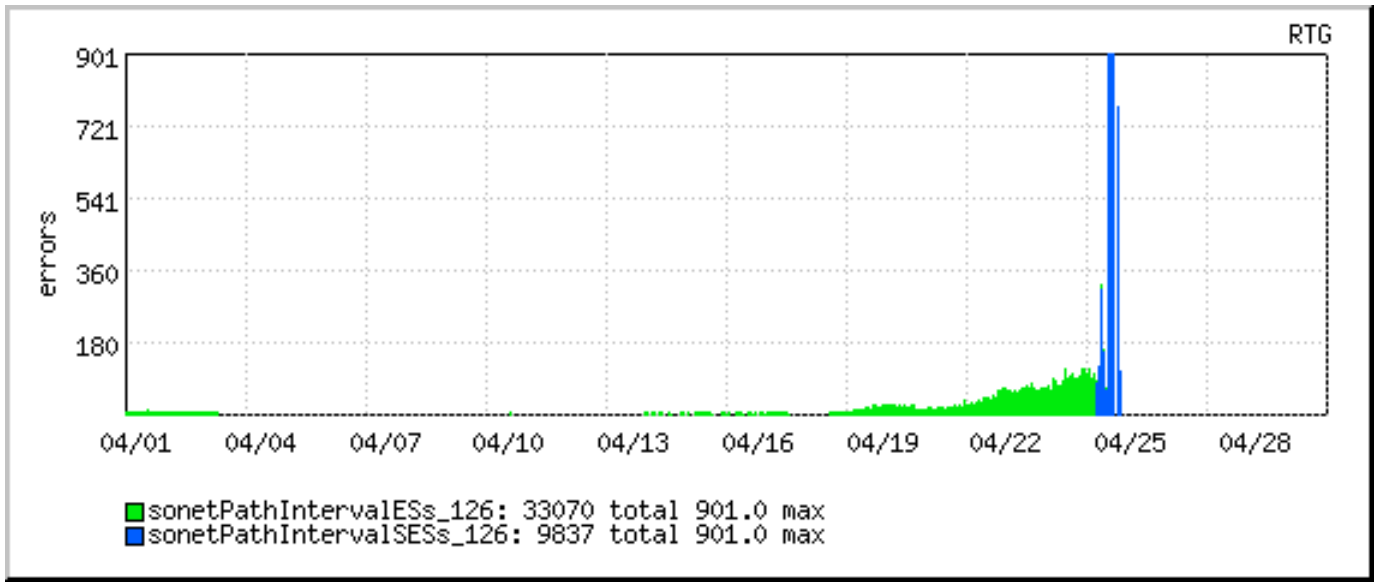
RTG Sub-Minute-Polling





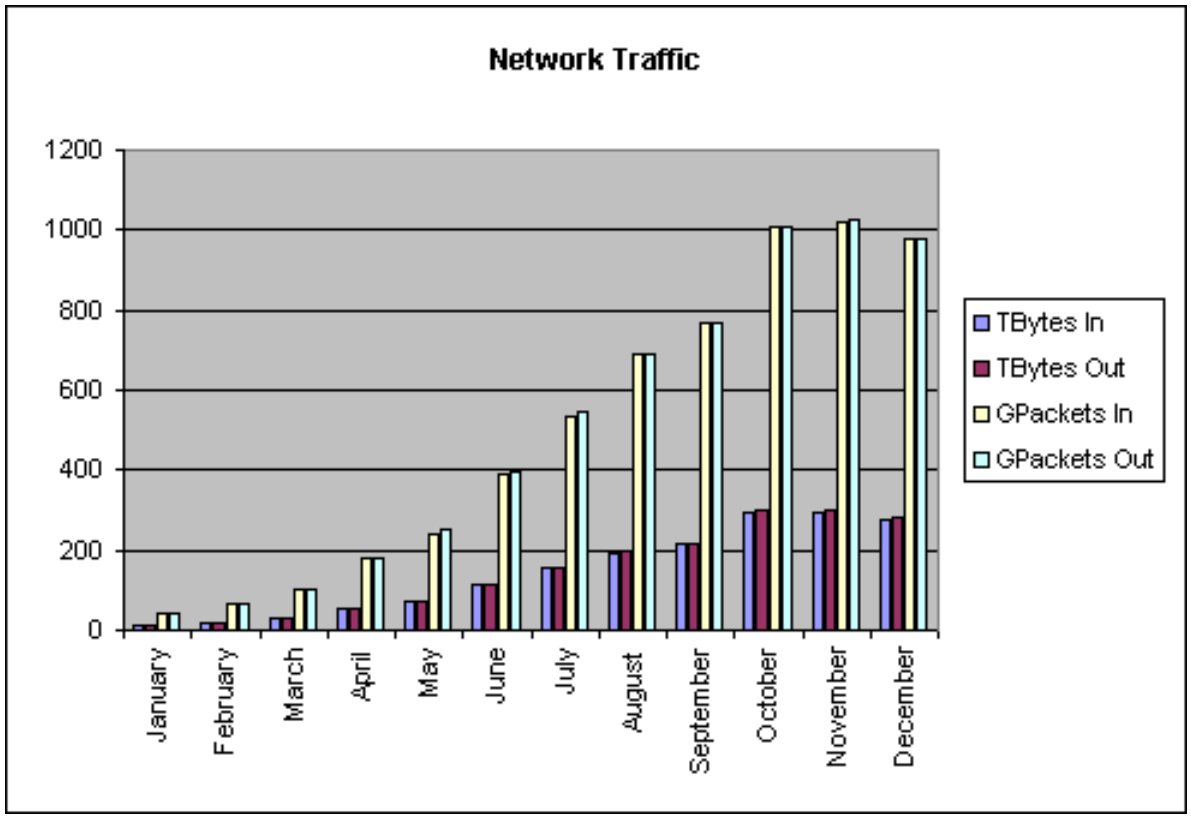
RTG Error Graph

- rtgplot can plot impulses (errors)



Long-Term Trending

- Perl scripts analyze data and produce CSV output that is easily imported into spreadsheets
- Ideal for management reports, trending, etc.



Thanks

- Questions?

RTG Home Page: http://rtg.sf.net