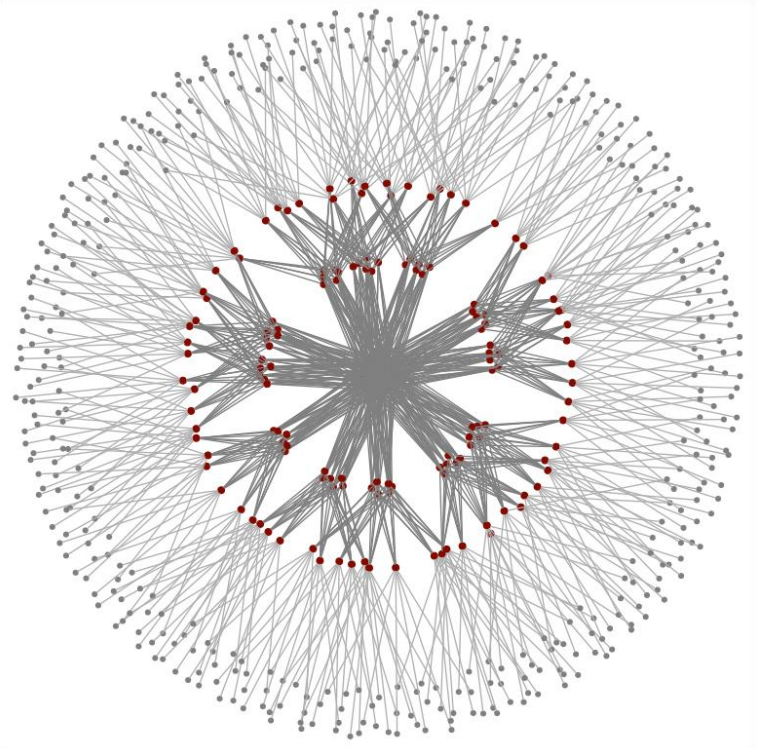
# Hawkeye

## Performance Monitoring for Data Centers

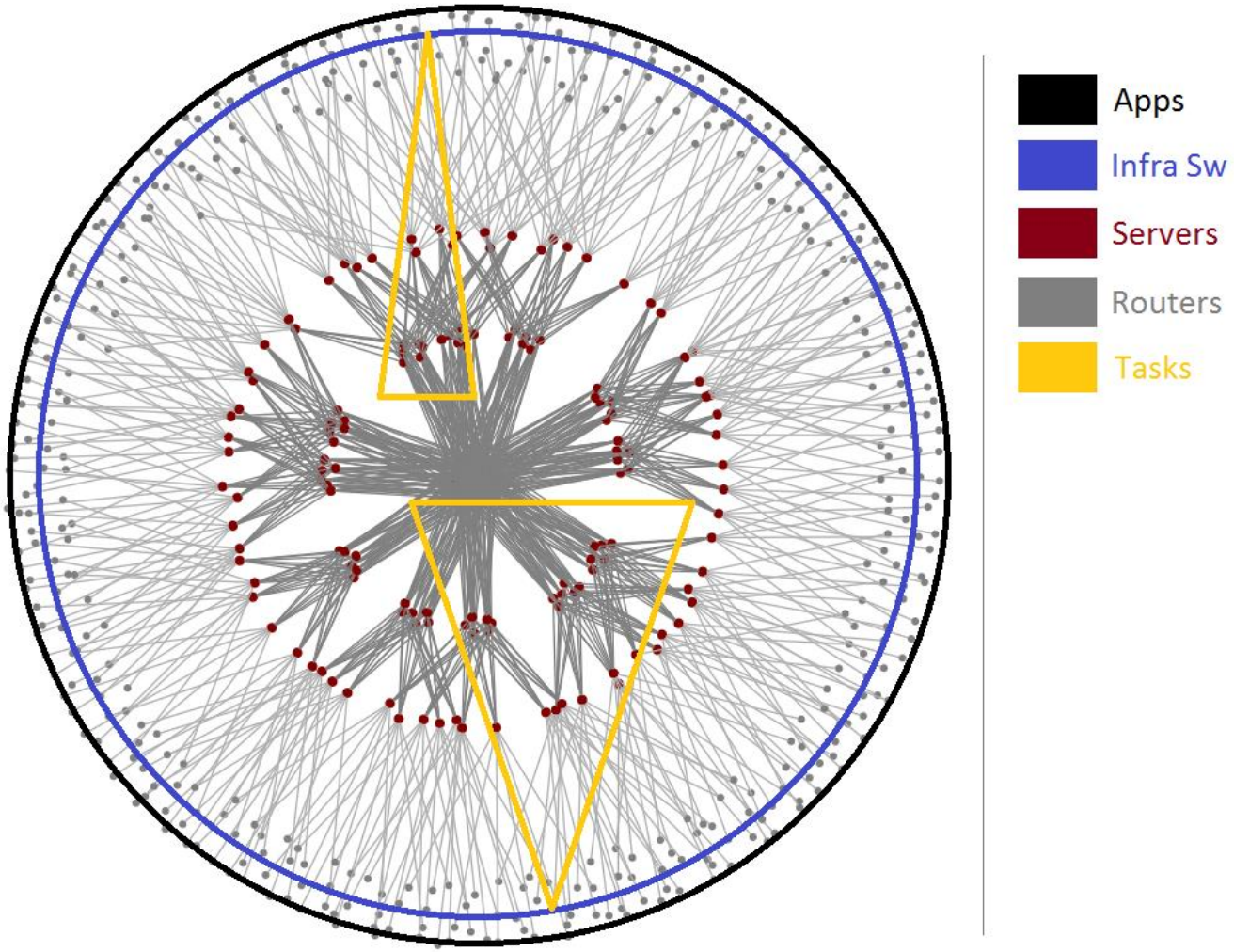


Shabbir Suterwala
Insight DE Fellow

# Hawkeye

- High throughput TCPIP packet sniffer
- Maps packet ID to App & Hw Stack
  - Stack Level → Hawkeye Monitors
- Measure packet latency and aggregate over time

# Hawkeye Monitors

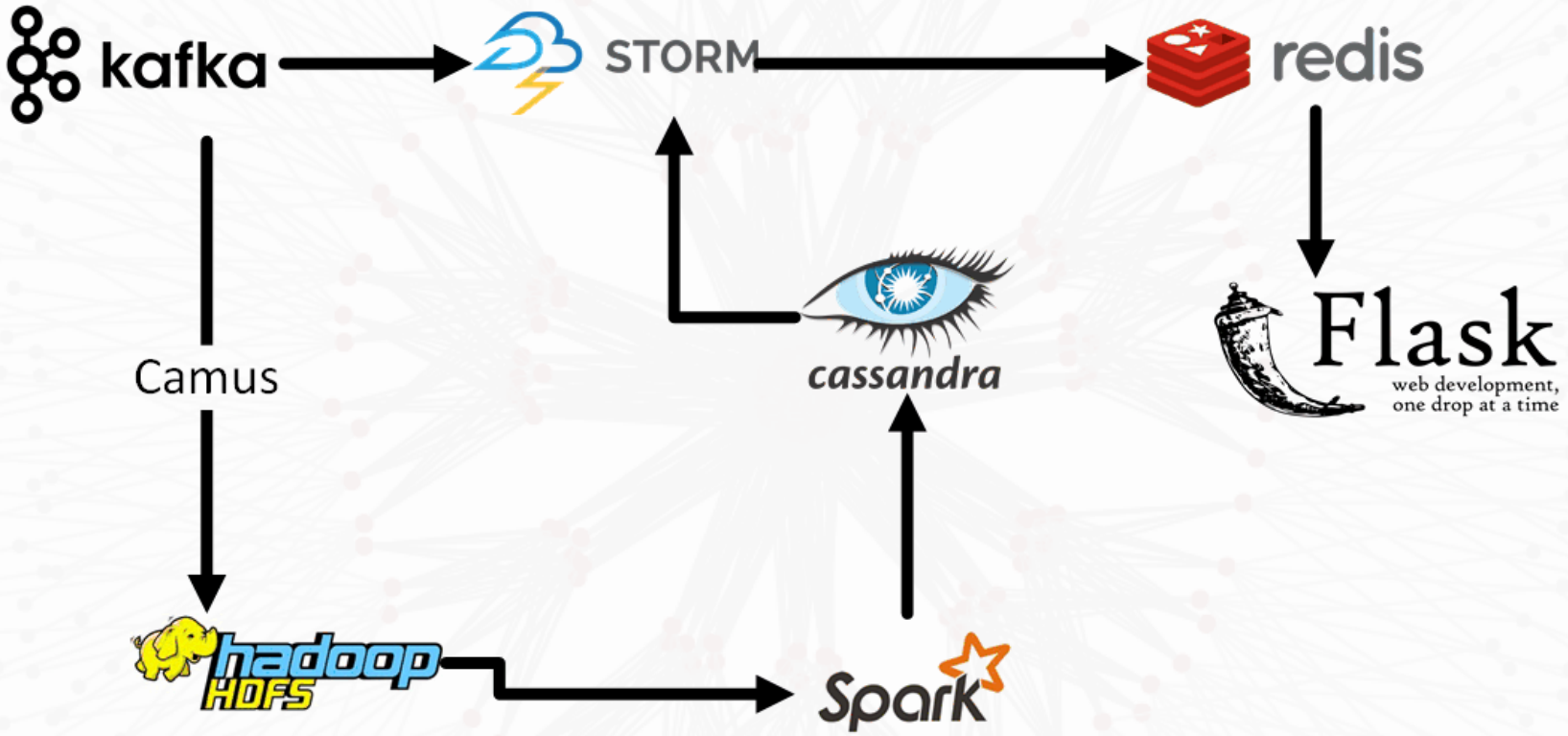


# Data

HawkeyeEvent: {
"TsIn": 1453407175613828, "TsOut": 1453407175614662,
~~"PacketID": "PACKET79217“,~~
“MonitorGroup": [
{type = “T”, subgroup = “AppType“, id = “App“},
{type = “I”, subgroup = " AppID“, id = "Hawkeye“},
{type = “T”, subgroup = "SwType“, id = "SWTYPE42“},
{type = “I”, subgroup = "SwID“, id = "SWID20“},
{type = “T”, subgroup = "TaskType“, id = "TASKTYPE217“},
{type = “I”, subgroup = "TaskID“, id = "TASKID154“},
{type = “T”, subgroup = "HwType“, id = "HWTYPE48“},
{type = “I”, subgroup = "HwID“, id = "HWID85730“},
{type = “T”, subgroup = "DevType“, id = "DEVTYPE86“},
{type = “I”, subgroup = "DevID“, id = "DEVID25438“},
]
}

Throughput : 125K messages / second

# Pipeline


kafka
STORM
redis
Camus
cassandra
Flask
web development,
one drop at a time
hadoop
HDFS
Spark

# Schema

```
cqlsh:hawkeye4> select * from monitor_alerts limit 10;

 monitor     | alert_time_year | alert_time_ms            | alert_sev | alert_through | max_through | min_through | sigma1neg_through | sigma1pos_through | sigma2neg_through | sigma2pos_through
-------------+-----------------+--------------------------+-----------+---------------+-------------+-------------+-------------------+-------------------+-------------------+-------------------
 TASKTYPE906 |            2016 | 2016-02-08 17:09:32+0000 |       red |   50332.85922 |  49526.7681 | 49337.22816 |       49297.97315 |       49566.02311 |       49163.94816 |        49700.0481
 TASKTYPE906 |            2016 | 2016-02-08 16:54:46+0000 |       red |   47908.37136 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:54:21+0000 |       red |       55228.96 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:54:01+0000 |       red |   50332.85922 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:53:18+0000 |       red |    49474.7233 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:53:01+0000 |       red |   44555.64286 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:52:25+0000 |       red |       49039.2 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:50:32+0000 |       red |     53526.275 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
 TASKTYPE906 |            2016 | 2016-02-08 16:48:21+0000 |       red |    49882.4733 | 49337.22816 | 49337.22816 |       49337.22816 |       49337.22816 |       49337.22816 |       49337.22816
   TASKID402 |            2016 | 2016-02-08 17:09:26+0000 |    yellow |   52517.13821 | 49344.80952 |  43740.5122 |       42579.82421 |        50505.4975 |        38616.98757 |        54468.33415

(10 rows)
cqlsh:hawkeye4> select * from monitor_history  limit 10;

 monitor     | record_time_year | record_time_ms           | nevents | tdeltaagg | time_window_size_ms
-------------+------------------+--------------------------+---------+-----------+---------------------
 TASKTYPE305 |             2016 | 2016-02-08 11:15:44+0000 |     400 |  20181033 |              600000
 TASKTYPE305 |             2016 | 2016-02-08 11:05:44+0000 |    1237 |  61621326 |              600000
 TASKTYPE305 |             2016 | 2016-02-08 10:55:44+0000 |    2813 | 139698082 |              600000
 TASKTYPE305 |             2016 | 2016-02-08 10:45:44+0000 |    1130 |  57024590 |              600000
 TASKTYPE305 |             2016 | 2016-02-08 10:35:43+0000 |    1868 |  91425185 |              600000
 TASKTYPE305 |             2016 | 2016-02-08 10:25:43+0000 |     213 |  10622745 |              600000
 TASKTYPE906 |             2016 | 2016-02-08 16:55:06+0000 |    2596 | 128571490 |              600000
 TASKTYPE906 |             2016 | 2016-02-08 16:45:06+0000 |    2060 | 101634690 |              600000
 TASKTYPE906 |             2016 | 2016-02-08 16:31:17+0000 |    1648 |  81307752 |              600000
 TASKTYPE906 |             2016 | 2016-02-08 16:21:17+0000 |    2060 | 101634690 |              600000

(10 rows)
```

# Cluster



10 m4.xlarge
1TB

Instances : $57.36 / 24 hours
Storage : $70 / month

# Lessons Learned

- Start with Query
  - Don’t start with performance modeling – Systems Engineering
  - Don’t start with data modeling – Relational Modeling
- Understand Query Execution
  - Cassandra crashed sometimes with Camus
- Use dashboard and tools for monitoring
  - nmon
- Tune later

# Shabbir Suterwala

- Team Lead, Architecture / Principal Architect @ Infor
  - Cloverleaf → Ingestion & processing engine for healthcare market
- Previously worked at Cisco, AMD and Storage Startup
  - Built OS, Virtual Machines, File Systems

# Backup Slides

# Hawkeye – How?

- Chatty App → Low Performance
- Measure by counting TCP/IP Packets
  - Throughput = Total Packets / Time
- Include Network Latency
  - Throughput = (Total Latency / Total Packets) / Time

# Hawkeye – When?

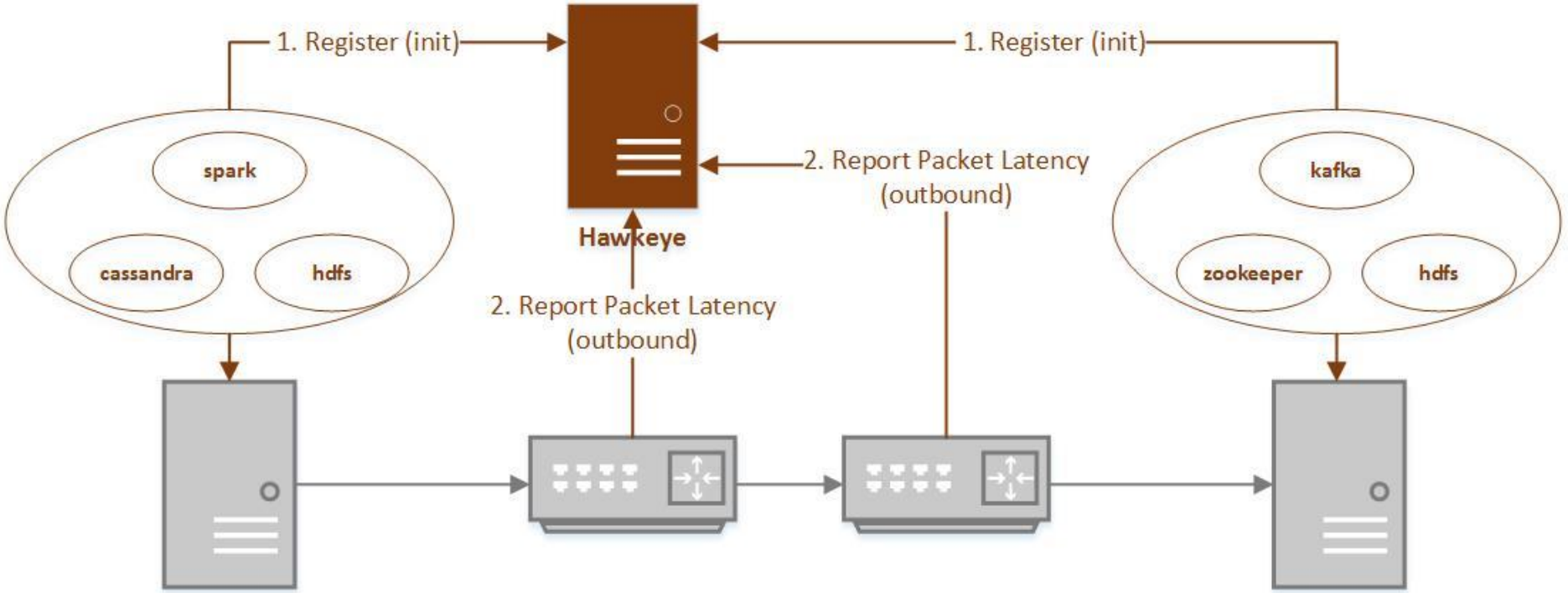
- Alerts – per monitor
  - Throughput now vs. historical data
  - Upper bound and lower bound
- Game – per stack
  - Rank user's stack's performance in real time

# Data

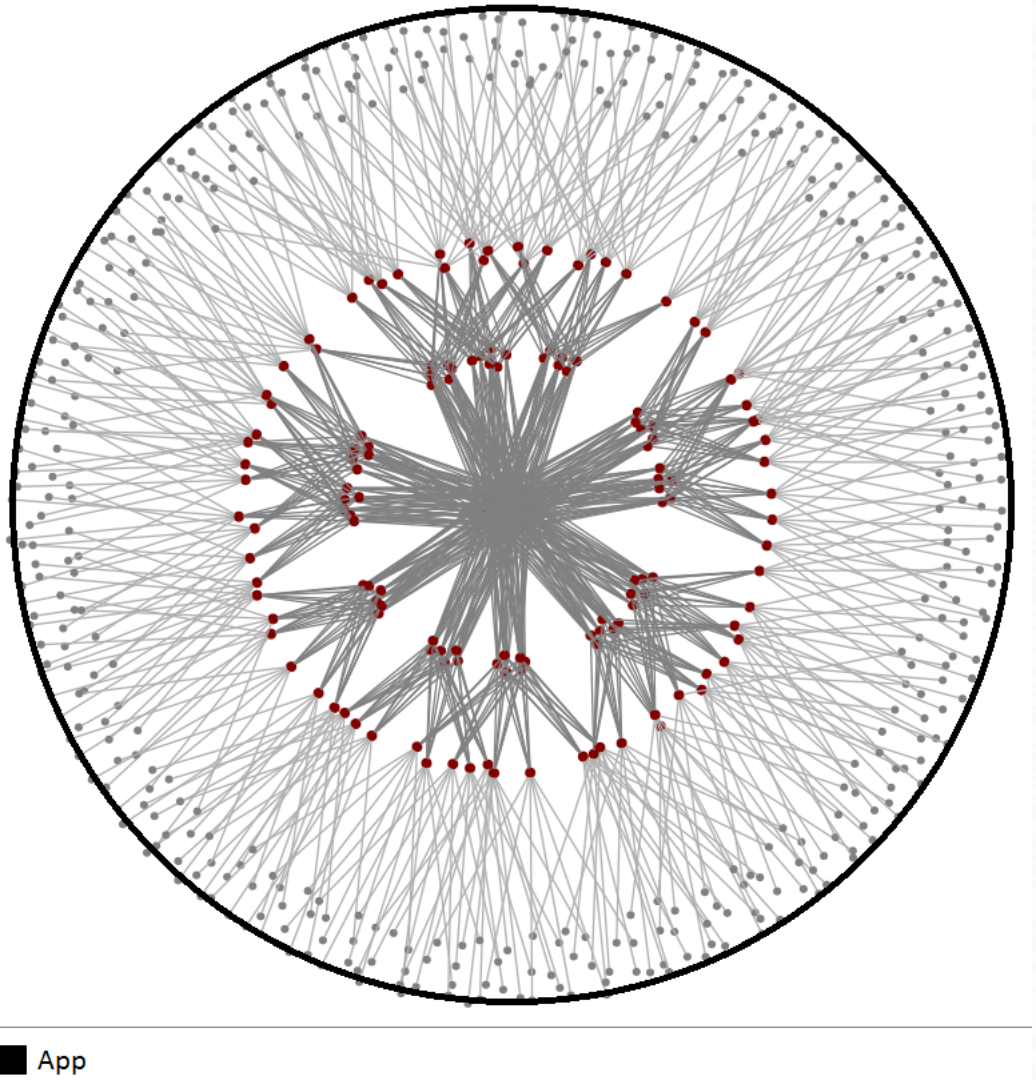
- Event:
  - TCP/IP packet timestamp In / Out
  - List of Monitors
- Engineered Data for Insight Project
- Real World
  - Hook into kernel: network stack, scheduler
  - Hi priority demon using *top*
  - Hypervisor

# Hawkeye - Intuition

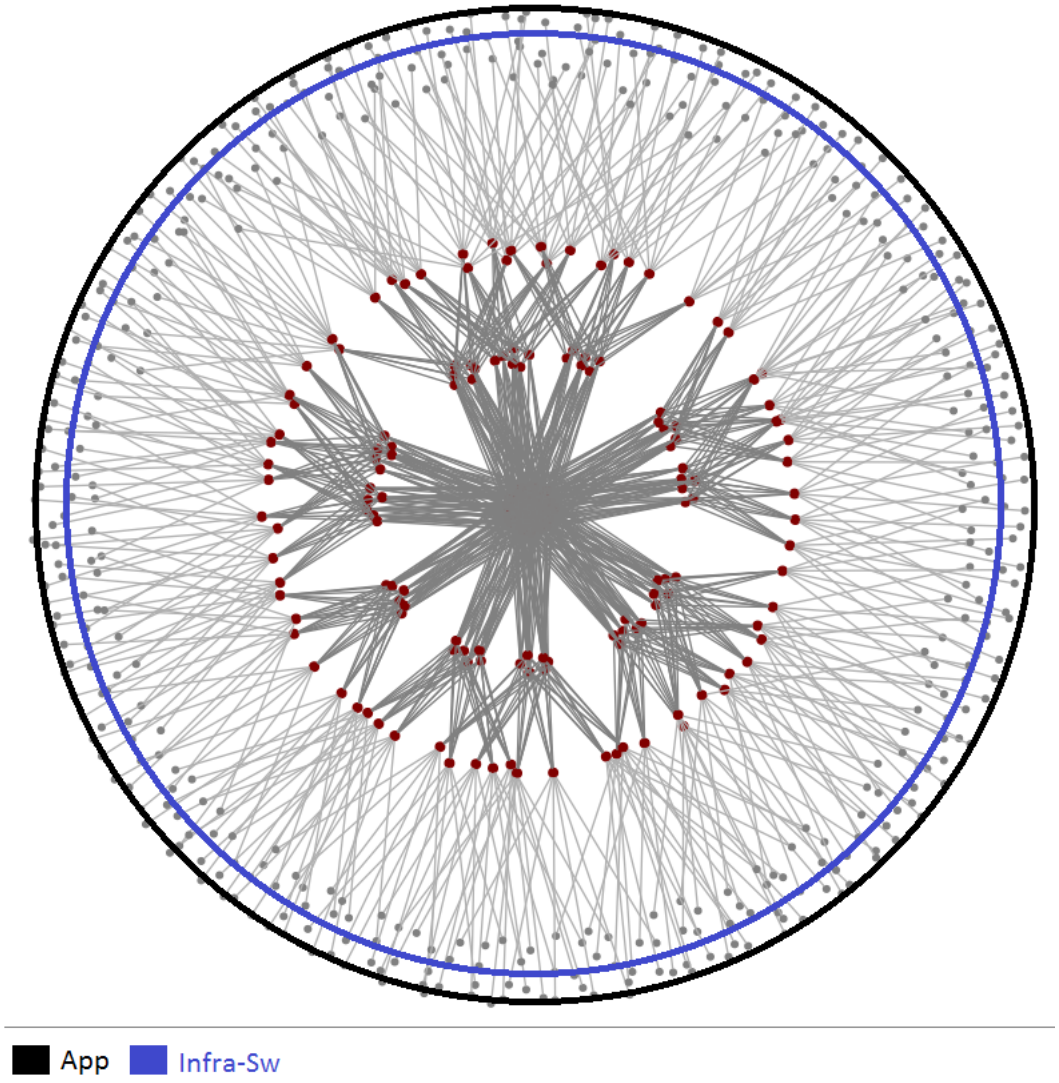
## Measure Network IO Total Latency



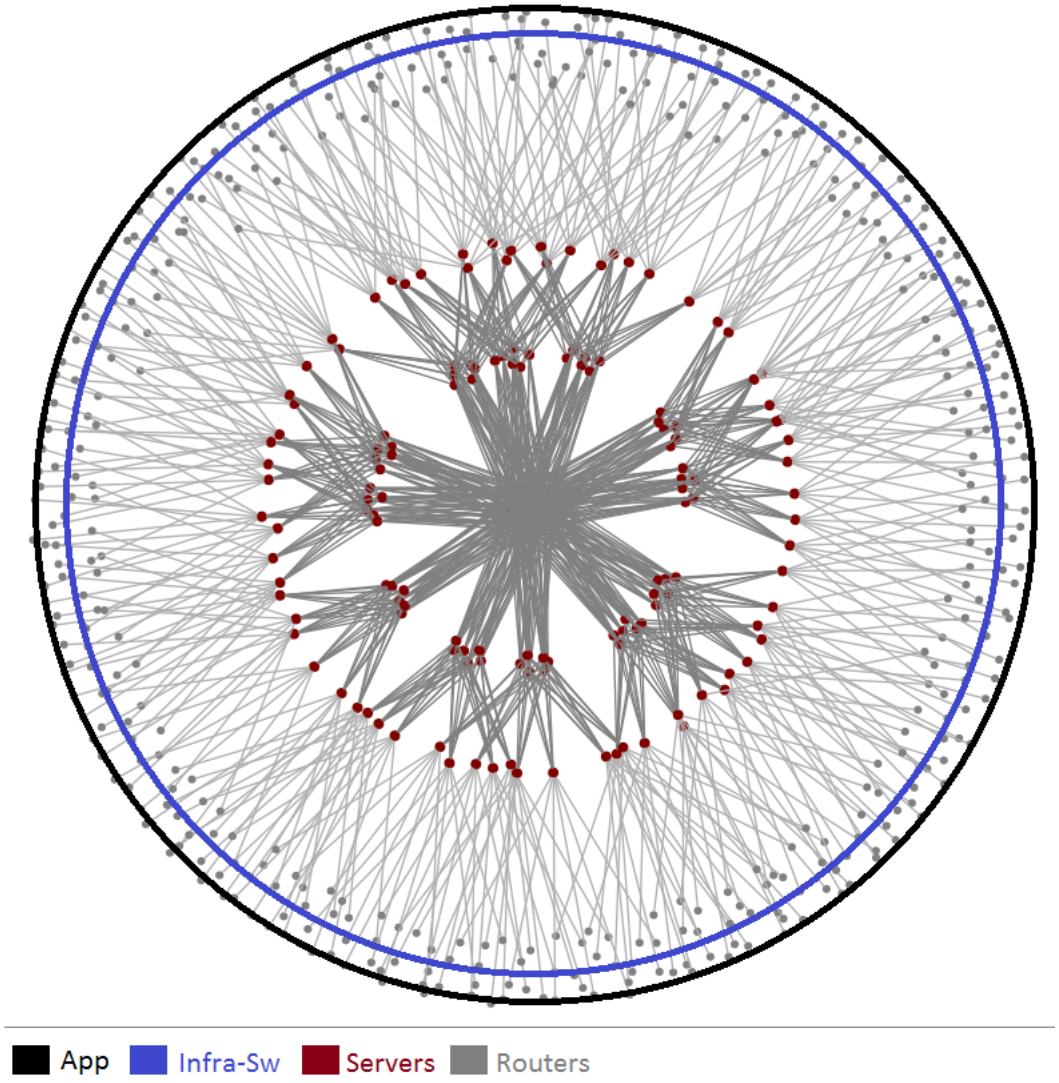
# Hawkeye Monitors



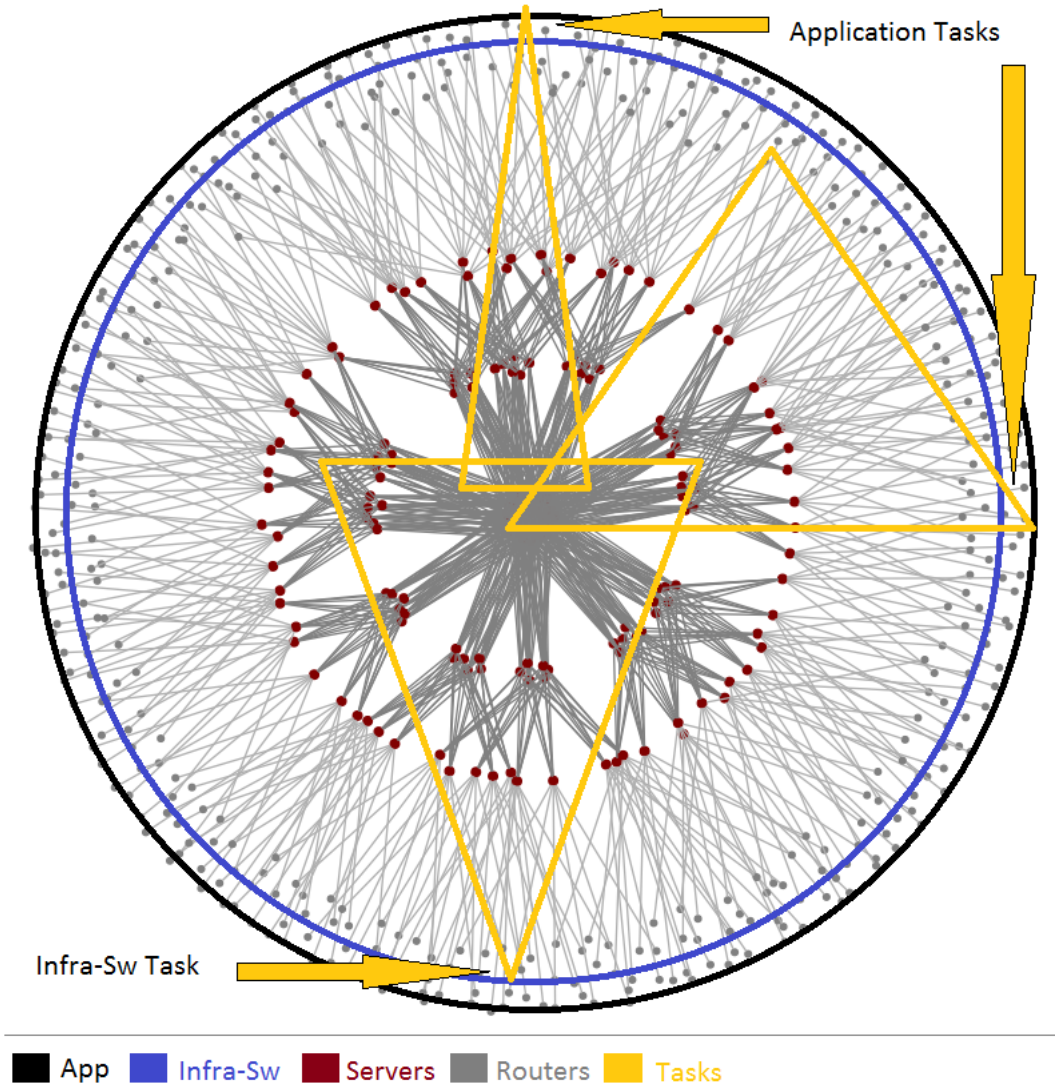
# Hawkeye Monitors (cont.)



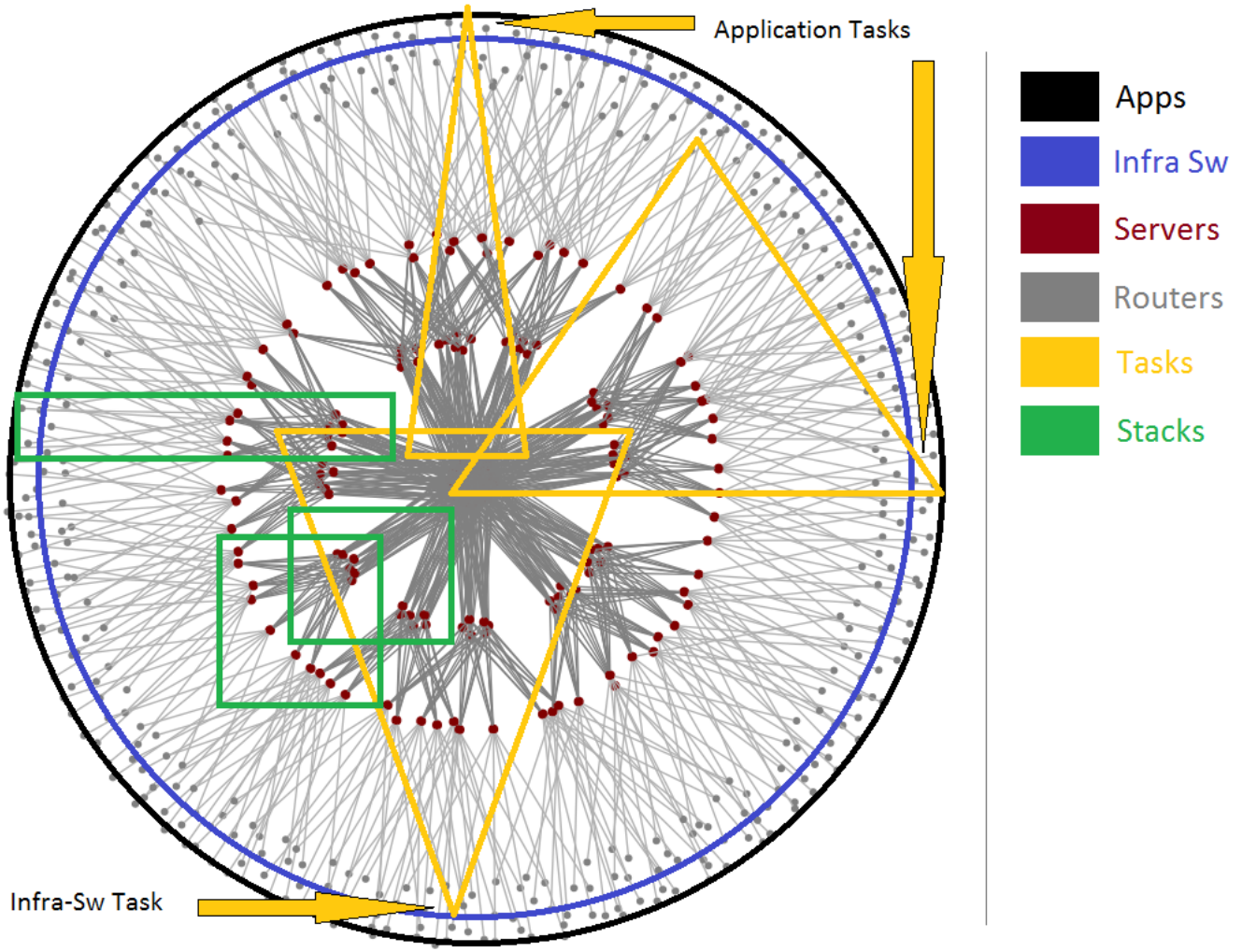
# Hawkeye Monitors (cont.)



# Hawkeye Monitors (cont.)



# Hawkeye Monitors



# Hawkeye – Execution