Advanced Monitoring

Introduction

- **Software Engineer at XS Software**
- **10 years of experience in everything development**
- **Mostly Web Development**

Epilogue

- **Thank you!**

What?

- Nothing revolutionary
- All big guys are doing it
- A lot of great tools are emerging currently
- Approach is more accessible now

What?

- A vision
- Work in progress
- Advanced approaches and techniques
- Needs implementation
- Focus on the principles

Assumptions

- **Targeted at small/medium sized infrastructure (100 - 150 hosts)**
- **Full stack development is appropriate for medium scale**
- **Developers are responsible for the production environment**
- **Web/HTTP based environments**

Why?

- **Small adoption**
- **Spread awareness of the approaches and emerging technologies in this field**

Traditional Approach to Monitoring

- Nagios based or similar
- Collectd/Graphite
- CPU, Memory, Network and Disk
- Process state
- Thresholds defined per host
- Alerts/Notification noise

Why monitor in the first place?

Why monitor in the first place?

- **Keep hosts up?**

Why monitor in the first place?

- Keep hosts up?
- Keep resource utilization low?

Why monitor in the first place?

- Keep hosts up?
- Keep resource utilization low?
- Keep our landing/index page up?

Why monitor in the first place?

- Keep hosts up?
- Keep resource utilization low?
- Keep our landing/index page up?
- Keep the application running?

Why monitor in the first place?

- **Support the business**
- **Keep the business running**
- **Make sure that the business value is delivered as intended**
- **Provide an overview of the technological environment**

Who are we monitoring for?

- **IT guys**

Who are we monitoring for?

- IT guys
- Pesky business guys

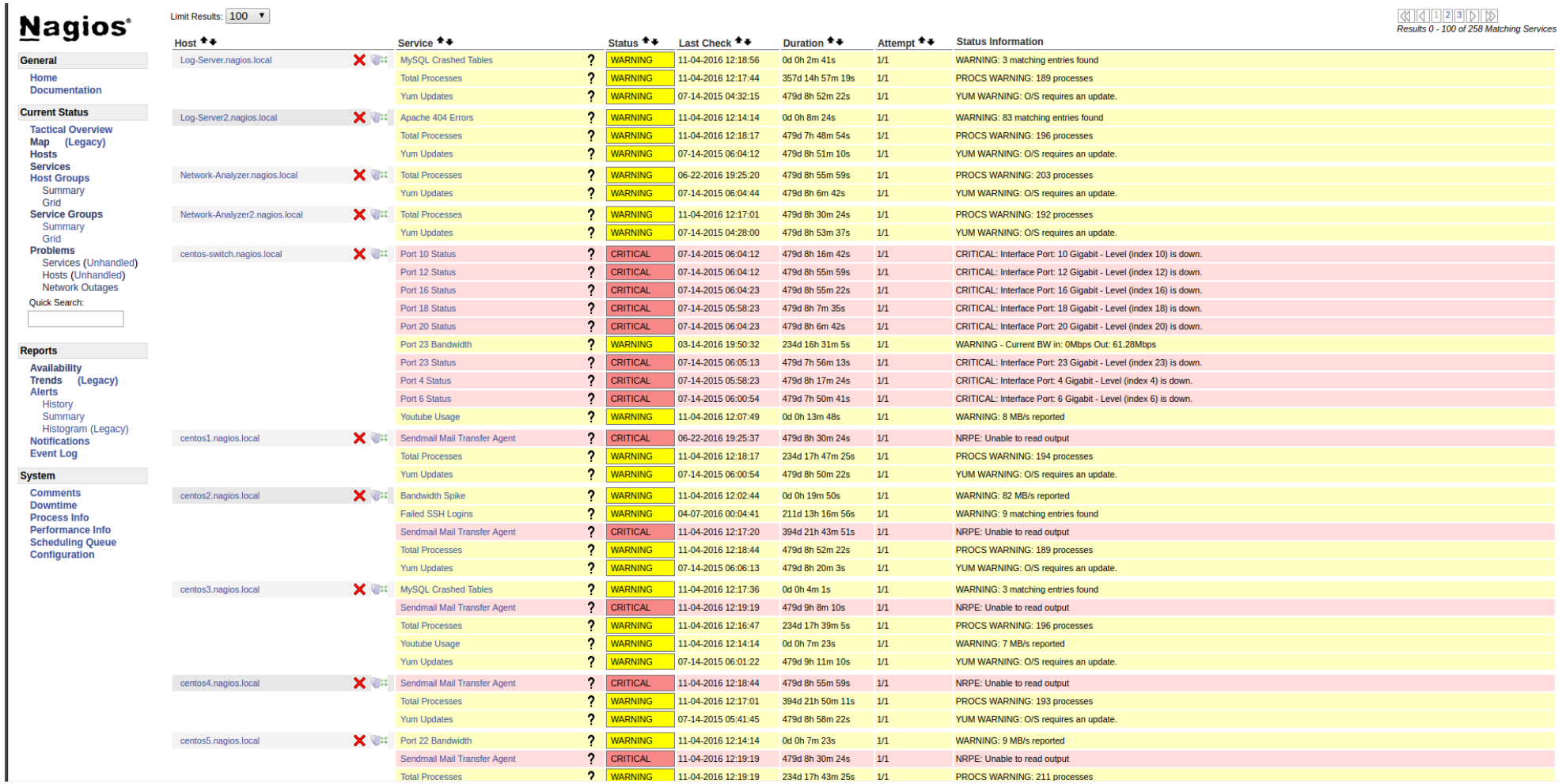
Traditional Approach Revisited

- Reactive
- Aimed at IT guys only
- Aimed at host resource utilization
- Aimed at keeping separate hosts/services running
- Tied to infrastructure

Traditional Approach Problems

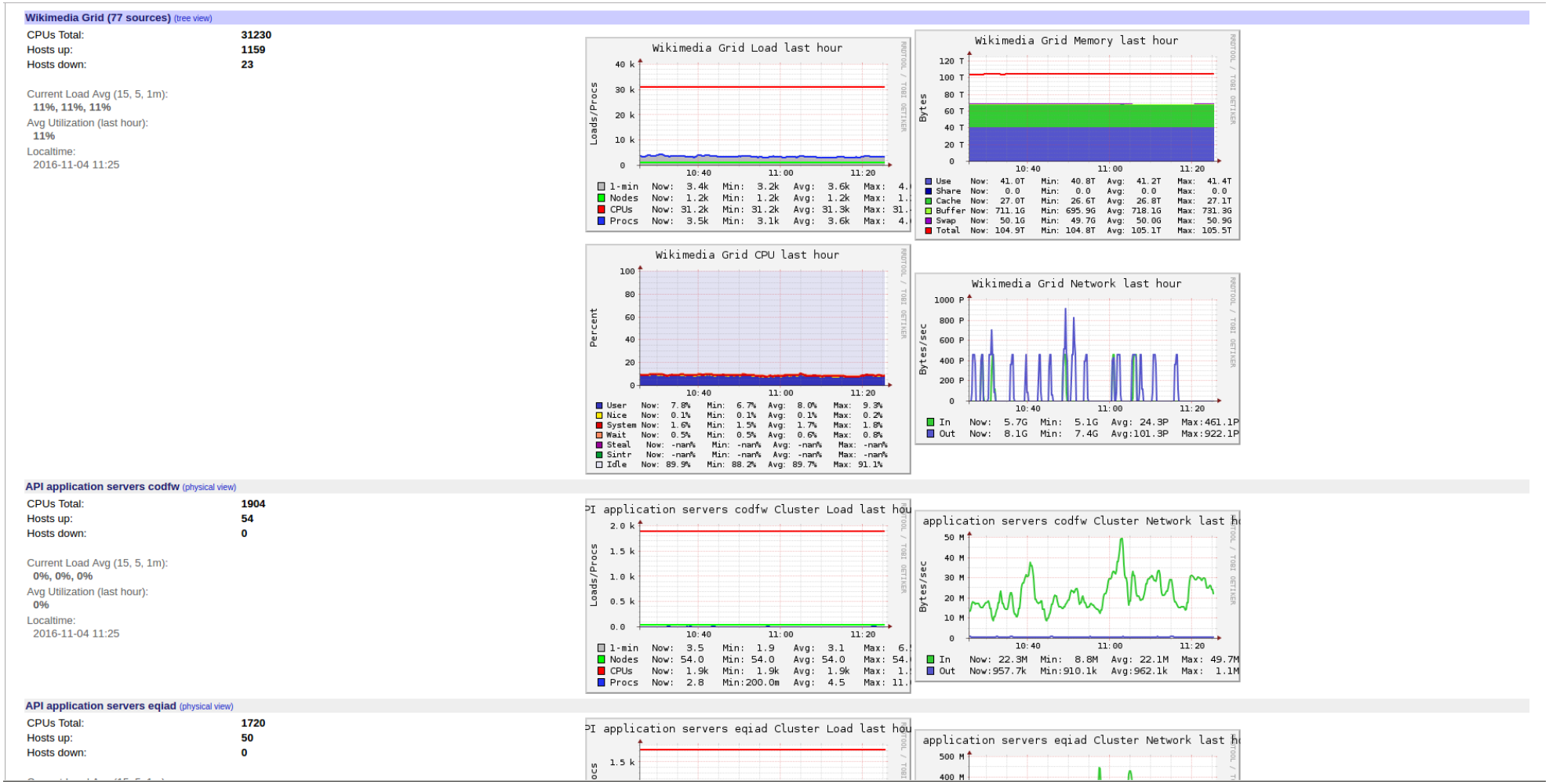
- Configuration management
- Threshold management
- Alerts/Notifications management
- Saturation/state checks

Traditional Approach Problems



Host	Service	Status	Last Check	Duration	Attempt	Status Information
Log-Server.nagios.local	MySQL Crashed Tables	WARNING	11-04-2016 12:18:56	0d 0h 2m 41s	1/1	WARNING: 3 matching entries found
	Total Processes	WARNING	11-04-2016 12:17:44	357d 14h 57m 19s	1/1	PROCS WARNING: 189 processes
	Yum Updates	WARNING	07-14-2015 04:32:15	479d 8h 52m 22s	1/1	YUM WARNING: O/S requires an update.
Log-Server2.nagios.local	Apache 404 Errors	WARNING	11-04-2016 12:14:14	0d 0h 8m 24s	1/1	WARNING: 83 matching entries found
	Total Processes	WARNING	11-04-2016 12:18:17	479d 7h 48m 54s	1/1	PROCS WARNING: 196 processes
	Yum Updates	WARNING	07-14-2015 06:04:12	479d 8h 51m 10s	1/1	YUM WARNING: O/S requires an update.
Network-Analyzer.nagios.local	Total Processes	WARNING	06-22-2016 19:25:20	479d 8h 55m 59s	1/1	PROCS WARNING: 203 processes
	Yum Updates	WARNING	07-14-2015 06:04:44	479d 8h 6m 42s	1/1	YUM WARNING: O/S requires an update.
Network-Analyzer2.nagios.local	Total Processes	WARNING	11-04-2016 12:17:01	479d 8h 30m 24s	1/1	PROCS WARNING: 192 processes
	Yum Updates	WARNING	07-14-2015 04:28:00	479d 8h 53m 37s	1/1	YUM WARNING: O/S requires an update.
centos-switch.nagios.local	Port 10 Status	CRITICAL	07-14-2015 06:04:12	479d 8h 16m 42s	1/1	CRITICAL: Interface Port: 10 Gigabit - Level (index 10) is down.
	Port 12 Status	CRITICAL	07-14-2015 06:04:12	479d 8h 55m 59s	1/1	CRITICAL: Interface Port: 12 Gigabit - Level (index 12) is down.
	Port 16 Status	CRITICAL	07-14-2015 06:04:23	479d 8h 55m 22s	1/1	CRITICAL: Interface Port: 16 Gigabit - Level (index 16) is down.
	Port 18 Status	CRITICAL	07-14-2015 05:58:23	479d 8h 7m 35s	1/1	CRITICAL: Interface Port: 18 Gigabit - Level (index 18) is down.
	Port 20 Status	CRITICAL	07-14-2015 06:04:23	479d 8h 6m 42s	1/1	CRITICAL: Interface Port: 20 Gigabit - Level (index 20) is down.
	Port 23 Bandwidth	WARNING	03-14-2016 19:50:32	234d 16h 31m 5s	1/1	WARNING - Current BW in: 0Mbps Out: 61.28Mbps
	Port 23 Status	CRITICAL	07-14-2015 06:05:13	479d 7h 56m 13s	1/1	CRITICAL: Interface Port: 23 Gigabit - Level (index 23) is down.
	Port 4 Status	CRITICAL	07-14-2015 05:58:23	479d 8h 17m 24s	1/1	CRITICAL: Interface Port: 4 Gigabit - Level (index 4) is down.
	Port 6 Status	CRITICAL	07-14-2015 06:00:54	479d 7h 50m 41s	1/1	CRITICAL: Interface Port: 6 Gigabit - Level (index 6) is down.
	Youtube Usage	WARNING	11-04-2016 12:07:49	0d 0h 13m 48s	1/1	WARNING: 8 MB/s reported
centos1.nagios.local	Sendmail Mail Transfer Agent	CRITICAL	06-22-2016 19:25:37	479d 8h 30m 24s	1/1	NRPE: Unable to read output
	Total Processes	WARNING	11-04-2016 12:18:17	234d 17h 47m 25s	1/1	PROCS WARNING: 194 processes
	Yum Updates	WARNING	07-14-2015 06:00:54	479d 8h 50m 22s	1/1	YUM WARNING: O/S requires an update.
centos2.nagios.local	Bandwidth Spike	WARNING	11-04-2016 12:02:44	0d 0h 19m 50s	1/1	WARNING: 82 MB/s reported
	Failed SSH Logins	WARNING	04-07-2016 00:04:41	211d 13h 16m 56s	1/1	WARNING: 9 matching entries found
	Sendmail Mail Transfer Agent	CRITICAL	11-04-2016 12:17:20	394d 21h 43m 51s	1/1	NRPE: Unable to read output
	Total Processes	WARNING	11-04-2016 12:18:44	479d 8h 52m 22s	1/1	PROCS WARNING: 189 processes
	Yum Updates	WARNING	07-14-2015 06:06:13	479d 8h 20m 3s	1/1	YUM WARNING: O/S requires an update.
centos3.nagios.local	MySQL Crashed Tables	WARNING	11-04-2016 12:17:36	0d 0h 4m 1s	1/1	WARNING: 3 matching entries found
	Sendmail Mail Transfer Agent	CRITICAL	11-04-2016 12:19:19	479d 9h 8m 10s	1/1	NRPE: Unable to read output
	Total Processes	WARNING	11-04-2016 12:16:47	234d 17h 39m 5s	1/1	PROCS WARNING: 196 processes
	Youtube Usage	WARNING	11-04-2016 12:14:14	0d 0h 7m 23s	1/1	WARNING: 7 MB/s reported
	Yum Updates	WARNING	07-14-2015 06:01:22	479d 9h 11m 10s	1/1	YUM WARNING: O/S requires an update.
centos4.nagios.local	Sendmail Mail Transfer Agent	CRITICAL	11-04-2016 12:18:44	479d 8h 55m 59s	1/1	NRPE: Unable to read output
	Total Processes	WARNING	11-04-2016 12:17:01	394d 21h 50m 11s	1/1	PROCS WARNING: 193 processes
	Yum Updates	WARNING	07-14-2015 05:41:45	479d 8h 58m 22s	1/1	YUM WARNING: O/S requires an update.
centos5.nagios.local	Port 22 Bandwidth	WARNING	11-04-2016 12:14:14	0d 0h 7m 23s	1/1	WARNING: 9 MB/s reported
	Sendmail Mail Transfer Agent	CRITICAL	11-04-2016 12:19:19	479d 8h 30m 24s	1/1	NRPE: Unable to read output
	Total Processes	WARNING	11-04-2016 12:19:19	234d 17h 43m 25s	1/1	PROCS WARNING: 211 processes

Traditional Approach Problems



Traditional Approach

- Actually not bad
- It is not about the tools
- It is an essential and required step
- Provides faster problem resolution
- Provides tools for diagnosing problem causes

Traditional stuff for the pesky guys approach

- Human automated monitoring
- 100% anomaly detection
- Fast resolution

How can we get better?

- **Proactive**
- **Focus on business value metrics**
- **Quality against availability**
- **Focus on changes instead of values**
- **Incorporate monitoring in the application design**

How can we get better?

- **Dashboards aimed at the business guys**
- **Minimize human monitoring**
- **Focus on data exploration**
- **Reduce alerts/notification noise**

How can we get better?

- **This is hard!**
- **This is an evolution of the standard approach**

Extended Metrics

- **Healthy UX Metrics**
  - # Logins
  - # Performance
  - # Support Tickets
  - # Twitter mentions (if you are big enough)
  - Any data that corresponds to the user experience

Extended Metrics

- **Traditional Business Metrics**
  - Revenue
  - Registrations
  - Transactions

Extended Metrics

- **Add the metrics from all of your architecture components**
  - Web servers
  - Databases
  - Caches
  - Crons
  - Upstream backends (Fastcgi, Passenger, etc.)

Extended Metrics

- **Metrics from the storage layer**
  - Example: catch cheaters in game applications (resource anomalies, rankings scores, etc.)
  - Example: game process queues

Extended Metrics

- **AERPU**

Extended Metrics

- **Average Error Rate Per User (NOT ARPU)**

Extended Metrics

- **How much is too much?**

Extended Metrics

- **Experiment!**

The Pareto Principle

- **80/20 Rule**
- **80% of the effects come from 20% of the causes**

Focus on Change Patterns

- **Alert on application data**
- **Monitor current data vs. historical data**
- **Monitor based on prediction algorithms**
- **Realtime anomaly detection**

Reduce alerts/notification noise

- **If it is not actionable stop alerting on it**
- **Alertception**

Focus on Data Exploration

- The Fast Feedback Loop (Bret Victor)
- Easy Graphing
- Keep historical data
- Monitor technical change events (code deployments, new server installations, software updates, etc.)

Problems

- **Shifted focus (infrastructure → application)**
- **Developers and DevOps, System Administration gap**
- **Data/events scaling**
- **Storage**

Problems

- **Application Architecture**
- **The observer effect**
- **Anomaly detection is very hard**

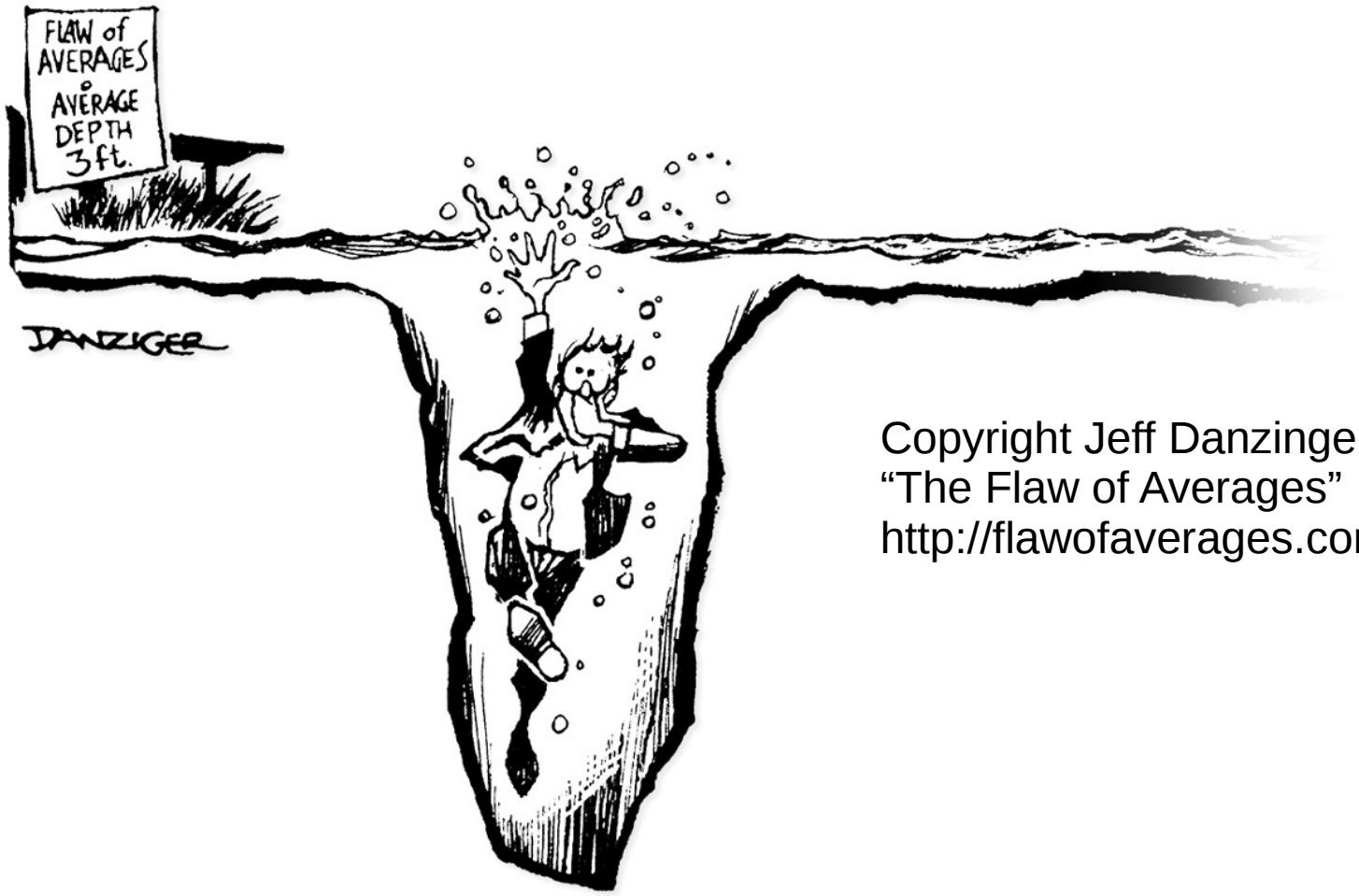
Implementation

- **Some Statistics**
- **Tools**
- **Challenges**

Statistics

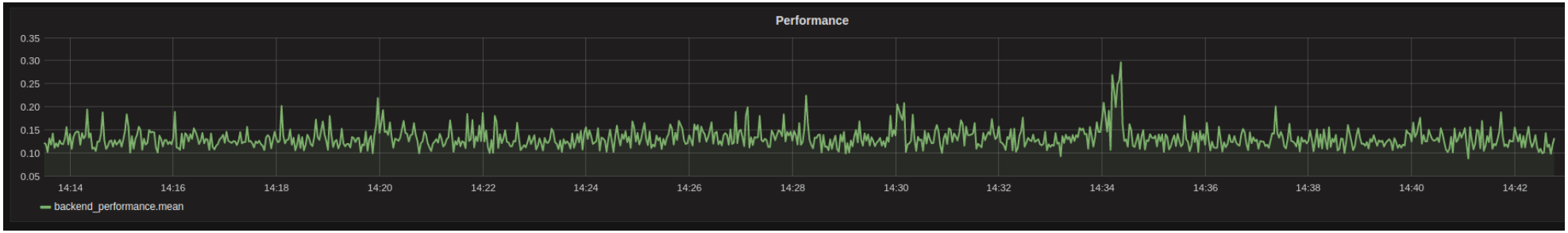
- **Averages**

Averages



Copyright Jeff Danzinger
“The Flaw of Averages”
http://flawofaverages.com/

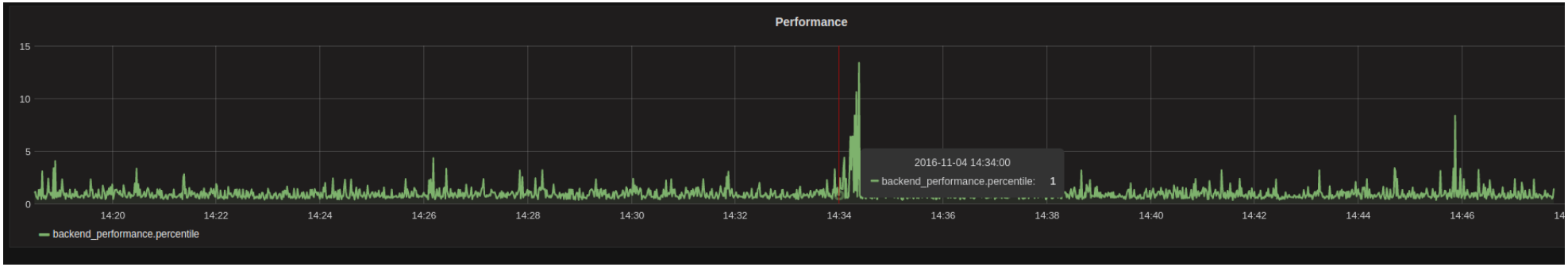
Averages


Performance
0.35
0.30
0.25
0.20
0.15
0.10
0.05
14:14
14:16
14:18
14:20
14:22
14:24
14:26
14:28
14:30
14:32
14:34
14:36
14:38
14:40
14:42
backend_performance.mean

Percentiles

- **A percentile (or a centile) is a measure used in statistics indicating the value below which a given percentage of observations in a group of observations fall. (Wikipedia)**
- **The distribution of values**

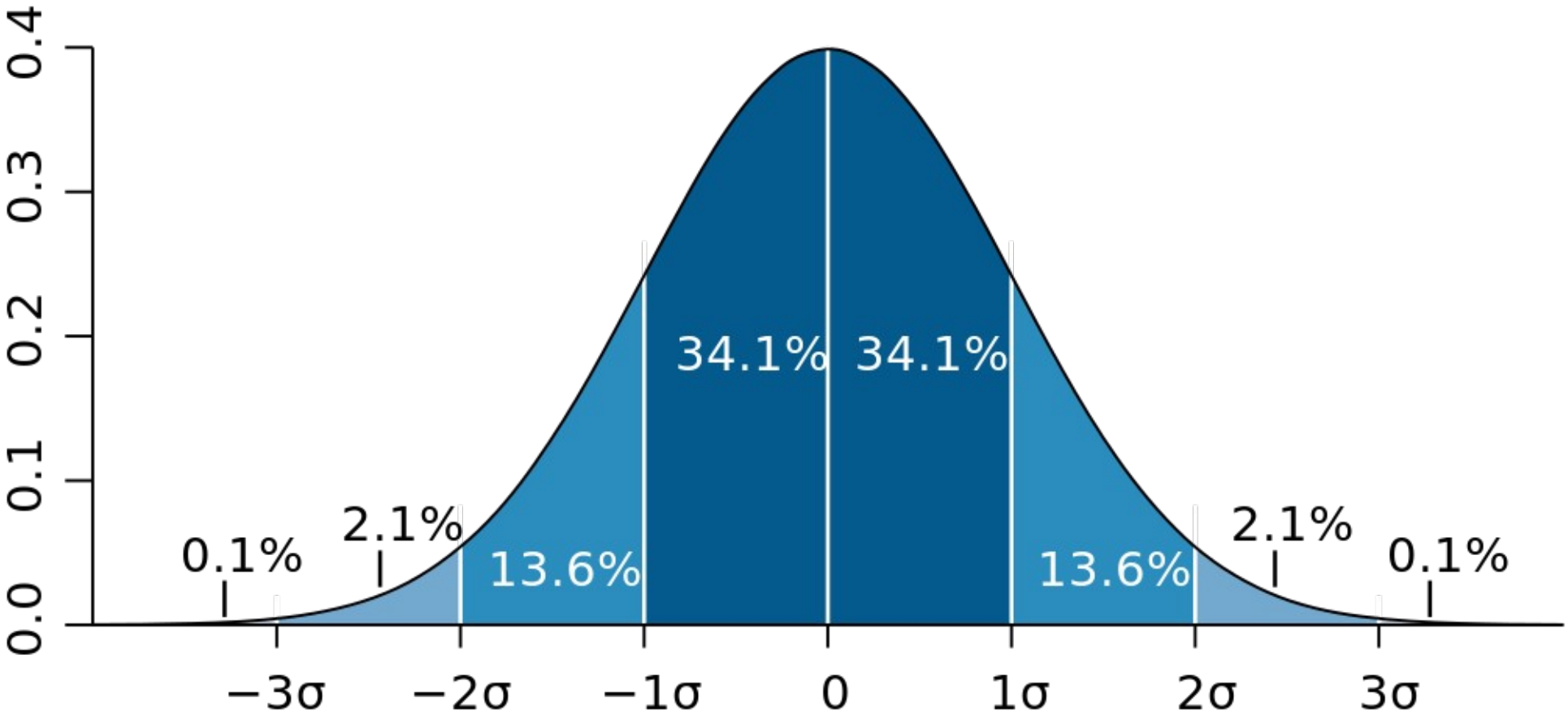
Percentiles



Standard Deviation

- **Measure for the variation of the data in a data set**
- **Low deviation → close to the mean (average)**
- **High deviation → far from the mean**

Normal Distribution



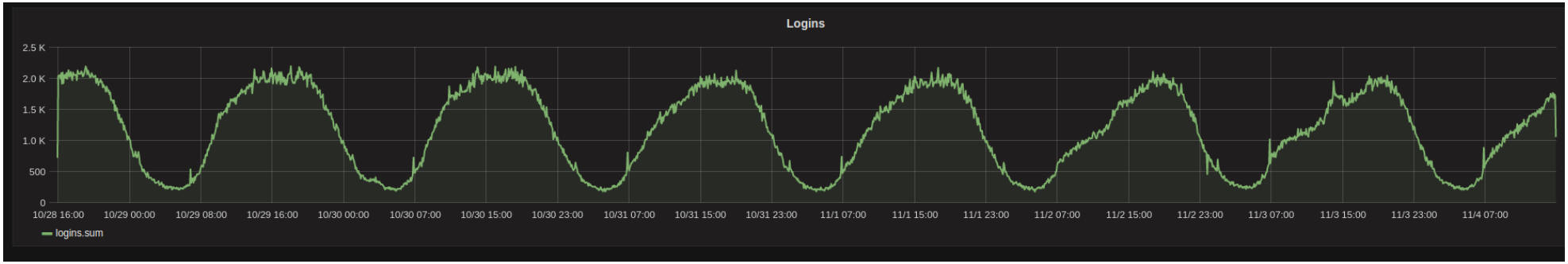
3 Sigma Rule

- **Anomaly Detection**
- **Check your distribution first!**

Holt-Winters Seasonal Method

- **Can be used for prediction of “seasonal data”**

Holt-Winters Seasonal Method



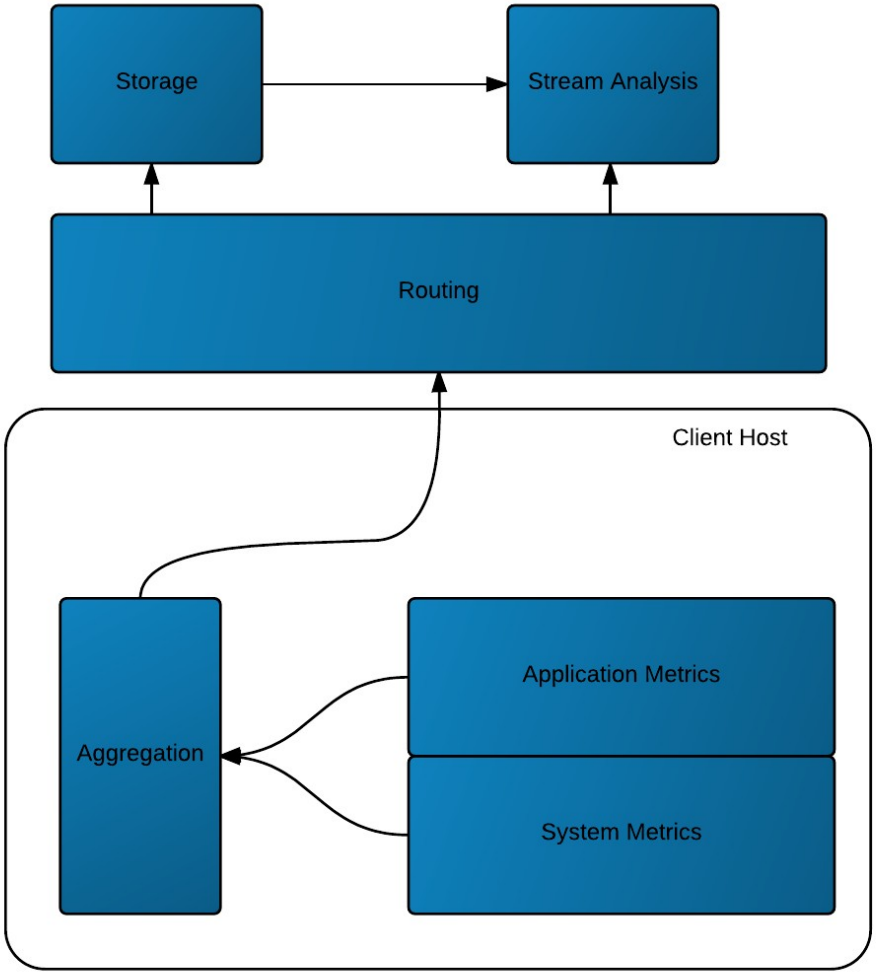
Tools

- A time series database
- A graphing solution
- Metric collection framework
- Event routing
- Tools for analyzing the stream of data

What is different?

- **Additional layer for event streams**
- **Timeseries databse allowing data exploration**
- **Easy interface for sending events from the application**

Architectural Overview



Tools

- InfluxDB
- Grafana
- Telegraf
- Custom Routing
- Kapacitor

InfluxDB

- Allows indexing events
- Provides good read/write speeds
- Provides statistical tools
- SQL like data exploration
- Collect dynamic metrics
- Active Development

Grafana

- Separation from data storage
- Provides awesome graphing and dashboards
- Active Community

Telegraf

- Metrics collection agent
- Supports batching
- Written in GO
- Good plugin architecture
- Collect dynamic metrics
- Collect system metrics
- Low overhead

Kapacitor

- Processes streams of data
- Works with InfluxDB seamlessly
- Statistical tools
- Alerting tools
- Can be used for alerting/anomaly detection
- Low overhead

Some Code

- **Application To Telegraf**

```
<?php
$point = $measurement . "," . implode(",", $tags) . " " .
implode(",", $fields) . PHP_EOL;
$result = @socket_write($this->socket, $point);
```

Telegraf → Router → InfluxDB (115 LOC)

```
// Create a point
pt, err := client.NewPoint(measurement, tags, fields, timestamp)

if err != nil {
    log.Fatalln("Error: ", err)
}

// Add point to batch if batch already exists
if bp, ok := pointBatches[MyDB]; ok {
    bp.AddPoint(pt)
} else {
    // Create a new batch
    bp, err := client.NewBatchPoints(client.BatchPointsConfig{
        Database:  MyDB,
        Precision: "ms", // use `u` for microseconds if needed
    })
    if err != nil {
        log.Fatalln("Error: ", err)
    }

    // Save batch to batch map for later use
    pointBatches[MyDB] = bp

    // Finally add point
    bp.AddPoint(pt)
}

// Write all batches
for _, bp := range pointBatches {
   err := c.Write(bp)
   if err != nil {
       log.Fatalln("Error writing: ", err)
   }
}
```

Kapacitor

- **Traditional Thresholds**

```
stream
    // Select just the cpu measurement from our example database.
    |from()
        .measurement('cpu')
    |alert()
        .crit(lambda: "usage_idle" < 70)
        // Whenever we get an alert write it to a file.
        .log('/tmp/alerts.log')
```

Kapacitor

- **3 Sigma**

```
stream
    |from()
        .measurement('cpu')
    |alert()
        // Compare values to running mean and standard
deviation
        .crit(lambda: sigma("usage_idle") > 3)
        .log('/tmp/alerts.log')
```

Kapacitor

- **Volume Change**

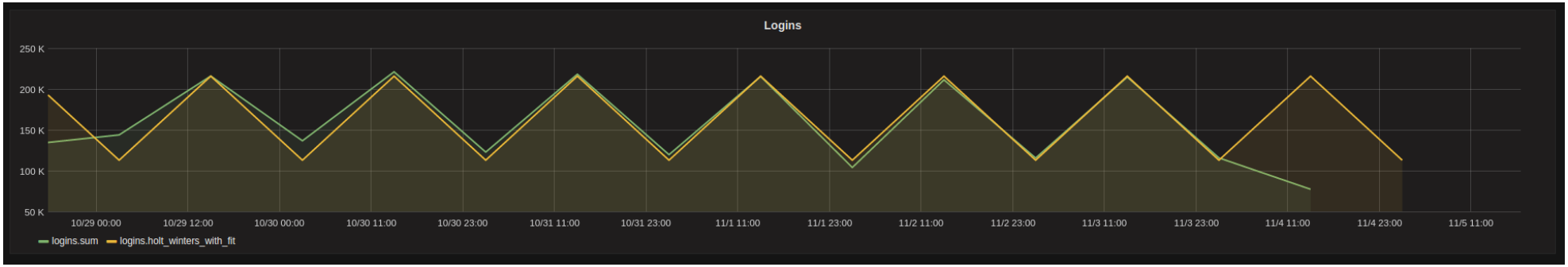
```
var current = batch
  |query('SELECT sum(count) as "logins" FROM "games"."default"."logins"')
    .period(15m)
    .every(15m)
    .align()

var historical = batch
  |query('SELECT sum(count) as "logins" FROM "games"."default"."logins"')
    .period(15m)
    .every(15m)
    .align()
    .offset(1d) //Yesterday
  |shift(1d)

current
  |join(historical)
    .as('current', 'historical')
    .tolerance(5s)
  |alert()
    .crit(lambda: "current.logins" / "historical.logins" < 0.7)
    .log('/tmp/historical_change.log')
```

Kapacitor

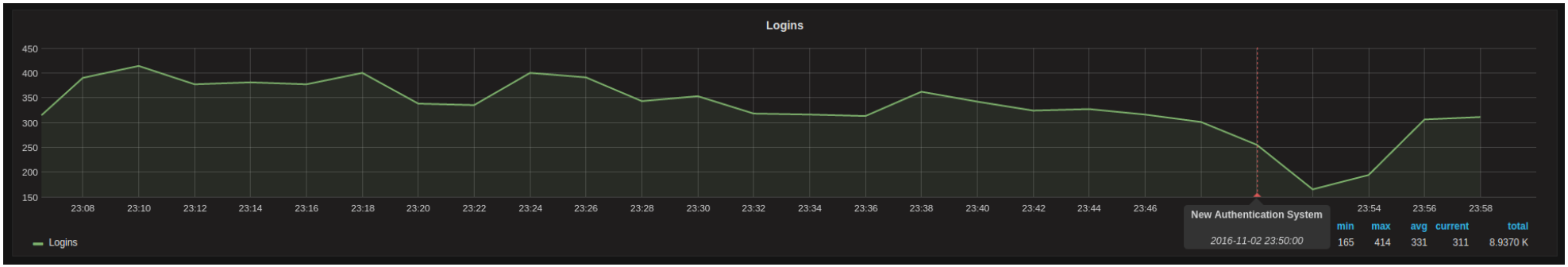
- **Slack Alert**

```
stream
  |from()
      .database('games')
      .measurement('logins')
  |window()
    .period(5m)
    .every(5m)
    .align()
  |sum('count')
    .as('logins')
  |alert()
      .crit(lambda: sigma("logins") >  3)
      .slack()
      .channel('#general')
      .log('/tmp/alerts.log')
```

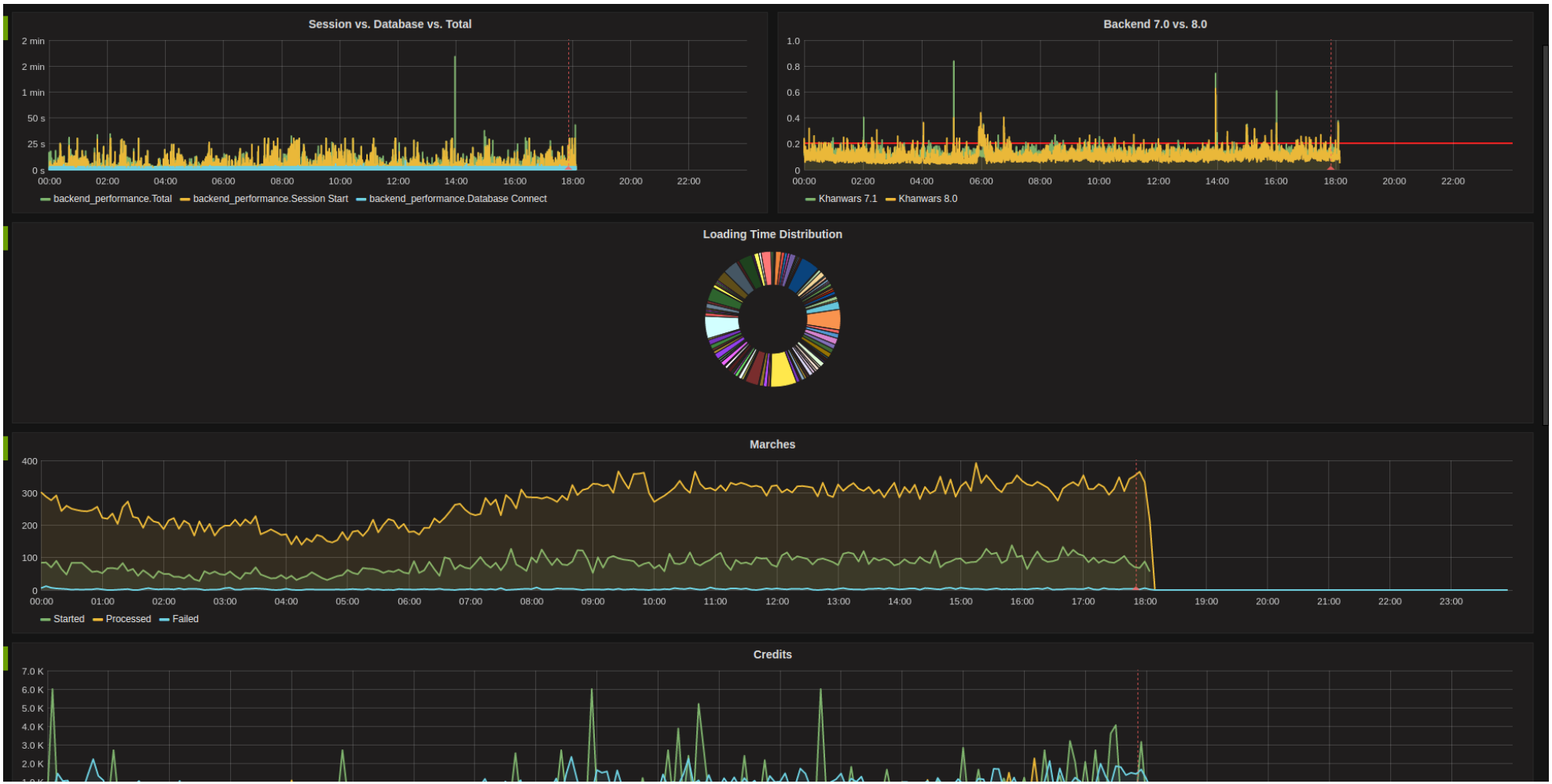
Grafana Prediction



Grafana Annotations



Grafana Dashboard



Weaknesses of the stack

- **Fairly young**
- **Problems with aggregating data**
- **Missing features**

Alternatives

- **Prometheus**
- **OpenTSDB**
- **Graphite (Whisper)**
- **KairosDB**
- **Riemann**
- **Chronograf (InfluxData)**
- **StatsD**
- **FluentD**

Possible Extensions

- **Better log aggregation and management**
- **Monitor the monitoring system**

The Future

- **Anomaly detection using machine learning**
- **Harddisk failure prediction using SMART stats**

References

- James Turnbull, “The Art of Monitoring”
- Preetam Jinka, Baron Schwartz, “Anomaly Detection for Monitoring”
- https://www.backblaze.com/blog/what-smart-stats-indicate-hard-drive-failures/
- http://www.kdd.org/kdd2016/papers/files/adf0849-botezatuA.pdf

Advanced Monitoring

- https://github.com/ablagoev
- alexander.i.blagoev@gmail.com
- Thank you!