



# Brought to You By



Marc Campbell



Martin Hald



inaka

Thanks Fernanado!

2013-03-20 15:49:06.243 [info] <0.2780.0>@tt\_account:came\_online:457  
Account came online: c2a8c653-d4c2-4d3d-ad2s-bd488cd9ec81/  
e5e752ef-79ae-4e26-9b6b-654e9926c56e

Latest Alerts

...

emergency a few seconds ago

The following queue "livestats\_queue" has exceeded the alert threshold of 50. Current item count of queue 163

67 recent

67 seen

Queue Length

Monitoring

NA

Alert

View Details

Clear Alert

ends ago at tt\_livestats line 187

uest body <<"message\_read">> [{message\_id,<<"297876932">>],[created\_time,<<"2013-03-

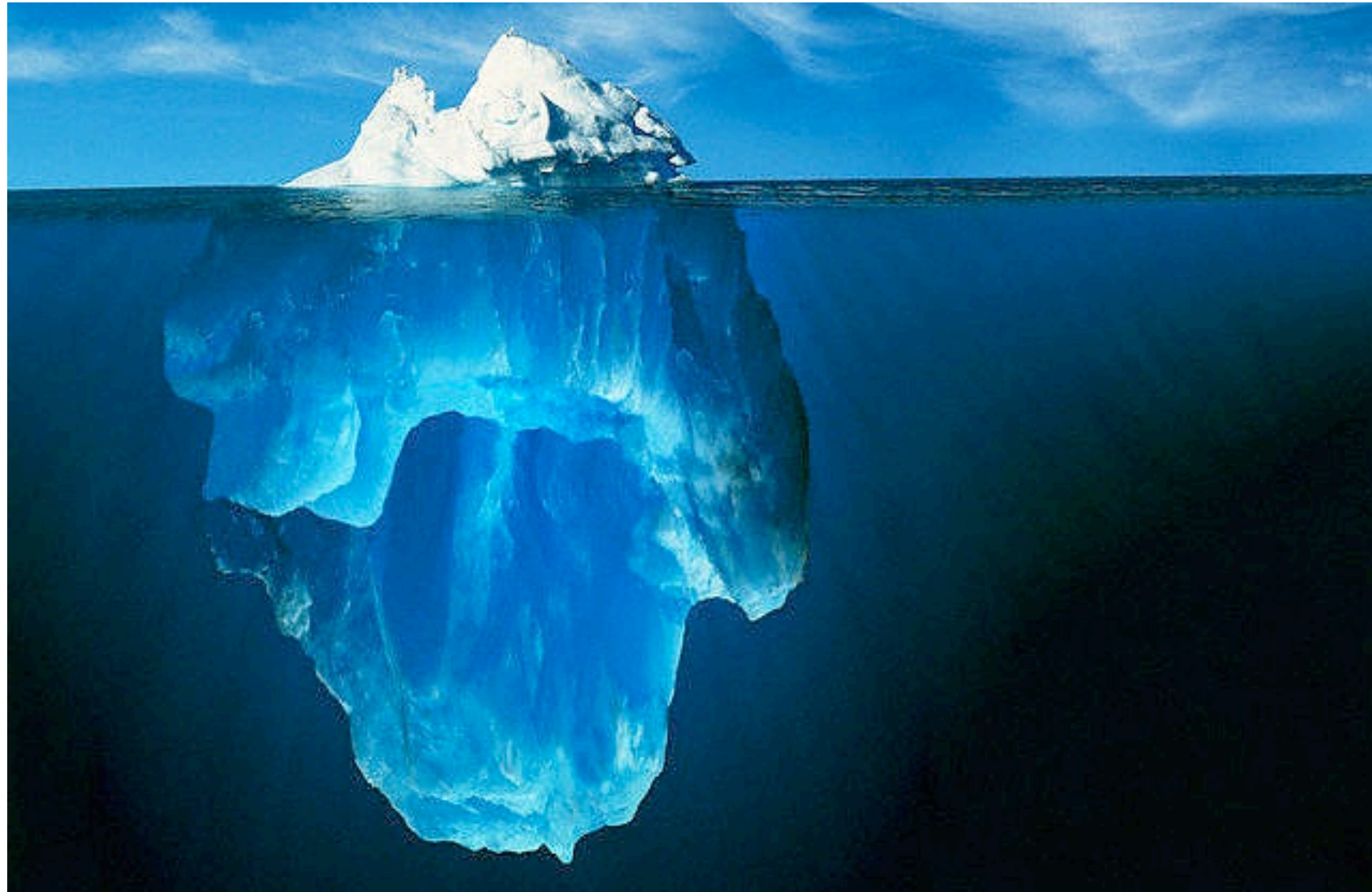
0Z">>],[product,<<"consumer">>],[organization\_token,undefined],[recipient\_resource,

<<"iPhone">>],[sender\_token <<"7fdh8hff-fcae-418e-b77f-a5daed940d78">>],[group false]]

# Live Demo



# Concepts



# Whats in a Log Entry?

## Lager

- Timestamp
- Severity
- Log Text
  
- Module
- Function
- Line Number
- Process ID

# Whats in a Log Entry?

## Lager

- Timestamp
- Severity
- Log Text
- Module
- Function
- Line Number
- Process ID

## Popcorn

- Role
- Node
- Version

## Coming Soon

- Caller Information
- App Metrics
- Env Metrics

# Lager Log Messages

```
88 2013-03-16 08:29:46.781 [info]  
<0.362.0>@avatar:req_handler:123 Starting  
avatar rendering
```

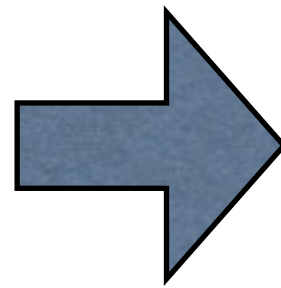


# Popcorn Log Message

```
88 2013-03-16 08:29:46.781 [info]  
<0.362.0>@avatar:req_handler:123 Starting  
#customer_request (avatar rendering) for  
@mhald
```

# From Log Entry to Alert

Log  
Message



Alert

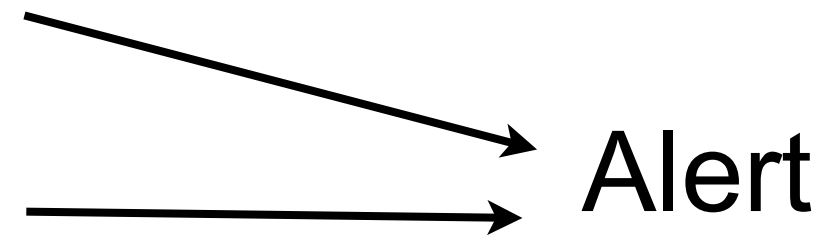
- Error
- Warning
- Critical
- Crash
- #identified\_topic

# Alerts Groups Critical Log Entries

#login\_failed for @fearing, returning #404

#login\_failed for @ga, returning #404

#login\_success for @bhatia, returning #200



# Address Scheme

Role	Version	Module	Line Number
<b>XMPP</b>	<b>v2.0.15</b>	<b>user_auth</b>	<b>: 54</b>

# Crashes have no line number

```
http_handler:404  
Bad request  
<<“error”>>
```

```
http_handler:404  
Bad request  
<<“api/v2”>>
```

```
http_handler:404  
Bad request  
*****
```

Auto  
Associative  
Memory

**0fc4cf26168b2a4b959d914682c10bc3**

# Data Stream

## Structured

- Role
- Node
- Version
- Severity

## Unstructured

- Identity
- Topic



# Acme Calculators



# Build Popcorn

```
git clone git@github.com:marccampbell/
```

```
popcorn.git
```

```
cd popcorn
```

```
./rebar get-deps
```

```
./rebar compile
```

```
lessc apps/popcorn/priv/css/popcorn.less > \  
      apps/popcorn/priv/css/popcorn.css
```

```
./rebar generate -f
```

# Configure Popcorn

## app.config

```
{http_auth_users, [{<<"admin">>, <<"password">>}]},  
{outbound_notifiers, [  
  {new_alert, outbound_mailer,  
    [{options, [{retries, 3}, {relay, "smtp.gmail.com"}],  
      {username, "acme.popcorn@gmail.com"},  
      {password, "PASSWORD"}]},  
    {from, "acme@logwithpopcorn.com"},  
    {recipients, ["your.email@domain.com]},  
    {subject, "New Alert at {{product}} {{version}}"}  
  ]},
```

# Hook in Popcorn

Calculator app rebar.config

```
{deps, [  
  ...  
  {lager_popcorn_backend, ".*", {git,  
    "git@github.com:marccampbell/  
    lager_popcorn_backend.git",  
    "master" }}  
  ...  
]}.
```

# Hook in Popcorn

## Calculator application sys.config

```
%% Lager config
{lager, [ {handlers, [
            {lager_console_backend, none},
            {lager_file_backend, [ {"log/error.log",
error, 104857600, "$D0", 5} ]},
            {lager_popcorn_backend, [
                                    {level,          debug},
                                    {popcorn_host,   "127.0.0.1"},
                                    {popcorn_port,   9125},
                                    {node_role,       "Calculator"},
                                    {node_version,   "1.0.7"}
                                ]}
        ]}
    ]}
```

# Demo

Using the Calculator HTTP API

```
curl -v -X POST --data-urlencode "2 4 /" \  
http://calculator.logwithpopcorn.com/api/calculator
```

1 5 +

9 3 /

5 2 - (will cause crash)

2 2 \*

See and try Popcorn at

<http://demo.logwithpopcorn.com/>



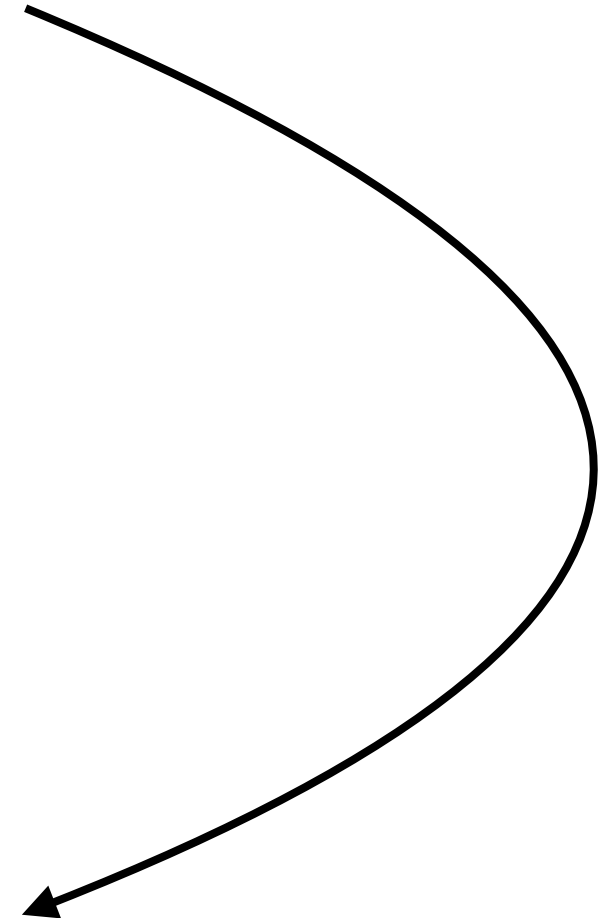


→ Customers

pingdom  
**Nagios®**



← Support



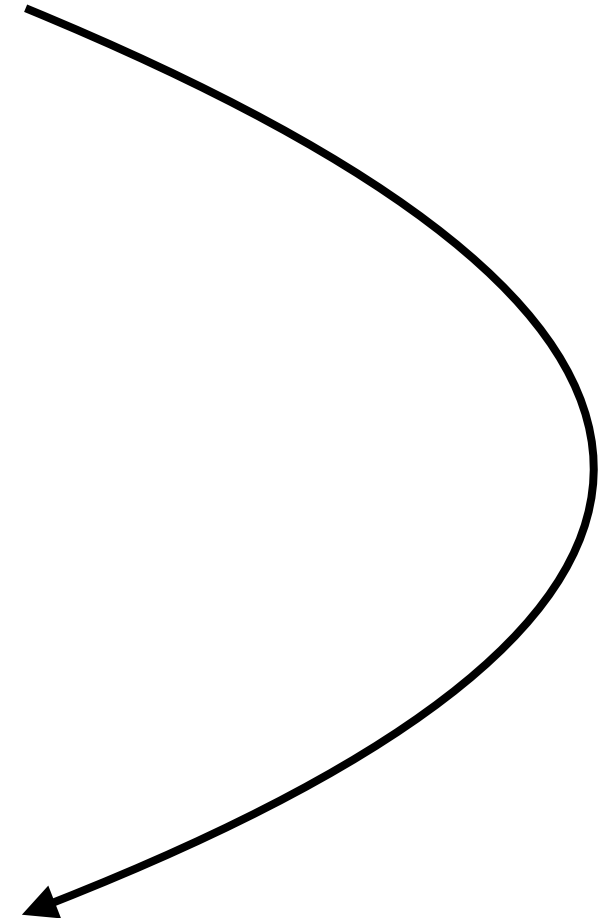


→ Customers

pingdom  
**Nagios®**



← Support





————→ Customers

- Identify edge case bugs
- Real-time alerting
- Logs as data streams

pingdom  
**Nagios®**



← Support



# Popcorn Design

**Ingesting**

**Storage**

**Rendering**

# UDP or TCP or ...

Connectionless

Quick

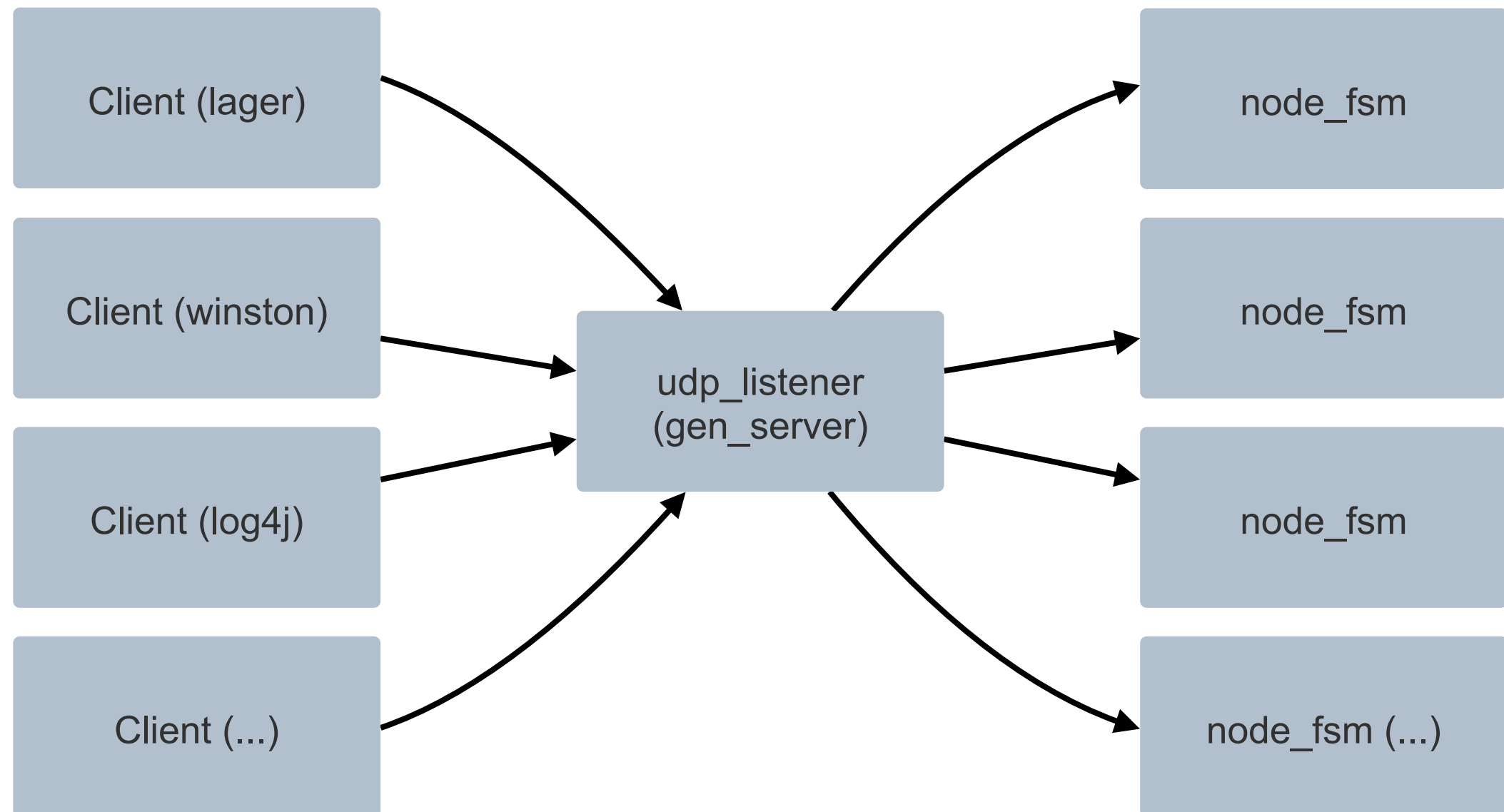
Out of order

Dropped packets

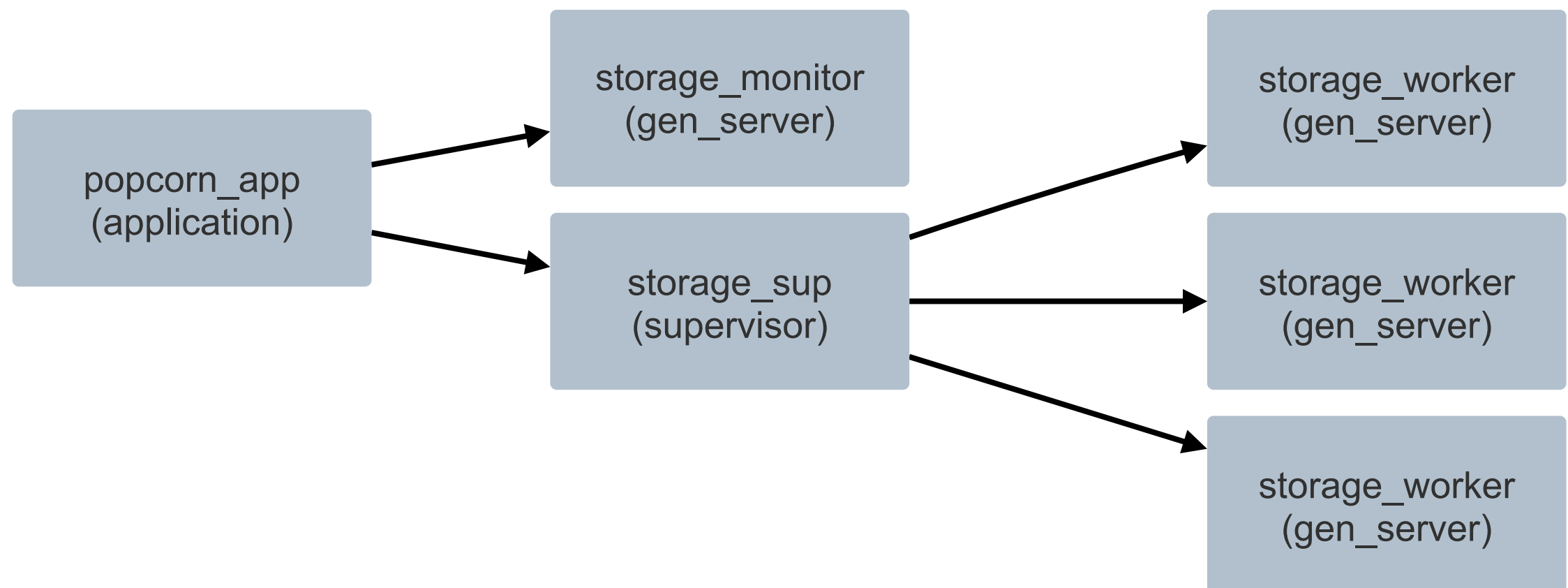
No  
acknowledgements



# Ingesting Architecture



# Storage Architecture



# Storage gen\_server behaviour

`init([]) / init([worker])`

`gen_server:handle_call/3`

`gen_server:handle_cast/2`

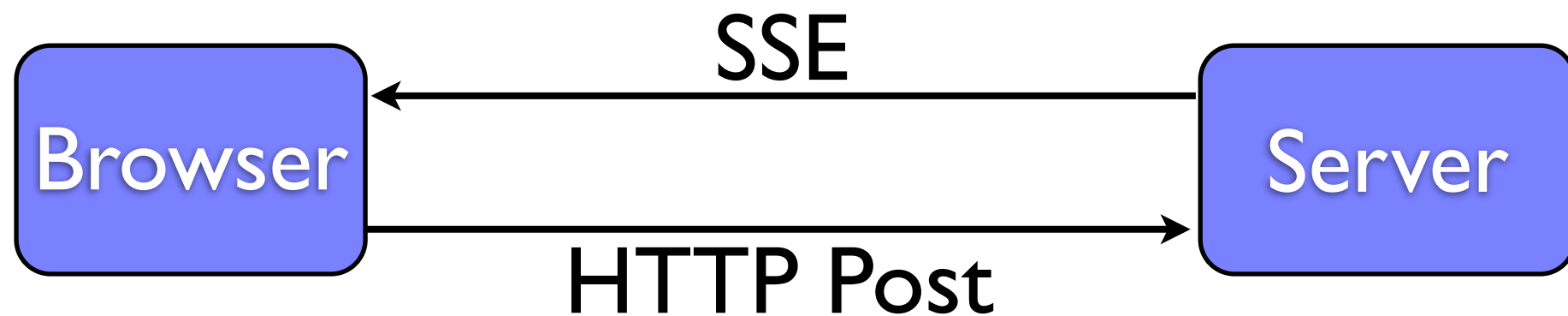
## handle\_call

```
{counter_value, Count}  
{is_known_node, Node_Name}  
{get_alert, Key}  
{get_alerts, Severity, Sort}  
{get_alert_keys, Type}  
{get_release_module_link, Role, Version, Module}  
{search_messages, {...}}  
{get_alert_timestamps, Severities}
```

## handle\_cast

```
{expire_logs_matching, Timestamp}
{send_recent_log_lines_matching, Pid, Count, Filters}
{new_log_message, #log_message}
{new_release_scm, Record}
{new_alert, Key, #alert}
{new_alert_timestamp, Key, Severity, #alert}
{new_alert_key, Type, Key}
{new_release_scm_mapping, Record}
{delete_counter, Counter}
{increment_counter, Counter, Increment_By}
{increment_counters, Counters}
{add_node, #popcorn_node}
```

# Server Sent Events



alert: {key: "001A", seen: 10}

node: {key: "xmpp@con8", log\_msgs: 324}



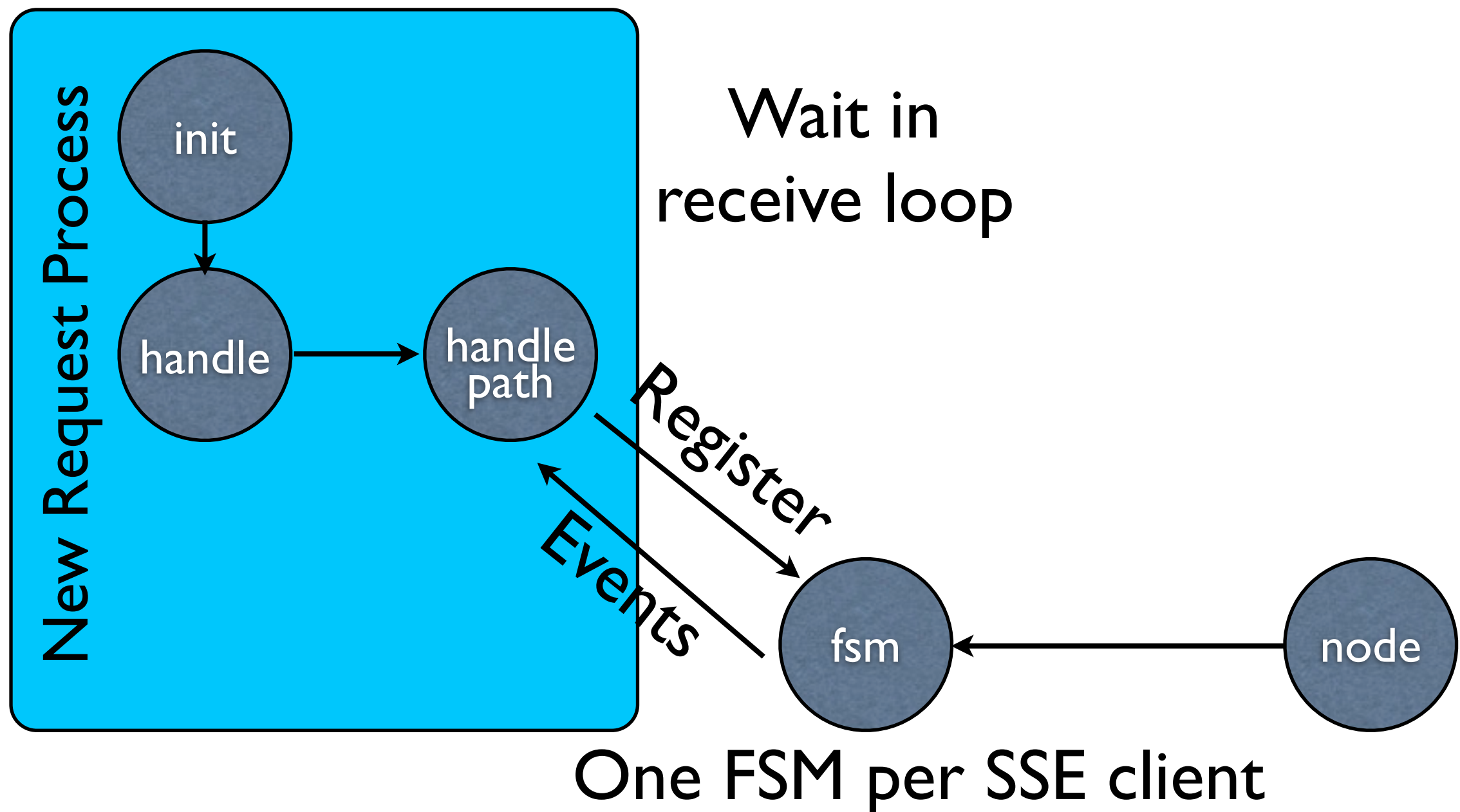
# SSE Events in JS

```
$(function() {  
    var streamSource = new EventSource("/dashboard/stream/{{stream_id}}");  
  
    streamSource.addEventListener(  
        'update_counters',  
        function(event) {  
            var counters = JSON.parse(event.data)  
            // ...  
        });  
  
    streamSource.addEventListener(  
        'new_node',  
        function(event) {  
            var counters = JSON.parse(event.data)  
            // ...  
        });  
});
```

# SSE Erlang Design

`http_stream_handler.erl`

`-behavior(cowboy_http_handler).`



```
server {  
    listen    80;  
    server_name popcorn.logwithpopcorn.com;  
  
    access_log /var/log/nginx/popcorn_access.log;  
    error_log /var/log/nginx/popcorn_error.log;  
  
    location / {  
        proxy_pass    http://localhost:9125;  
        proxy_redirect off;  
        proxy_buffering off;  
        proxy_cache off;  
        proxy_set_header Host $host;  
        proxy_set_header X-Real-IP $remote_addr;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_read_timeout 420;  
    }  
}
```

# Wire Protocol Standards & Efficiency

# Protobuffs

```
erlang:iolist_to_binary([
    protobuffs:encode(1, 1, uint32),    %% Packet version
    protobuffs:encode(2, atom_to_list(Node), string),
    protobuffs:encode(3, Node_Role, string),
    protobuffs:encode(4, Node_Version, string),
    protobuffs:encode(5, lager_util:level_to_num(Severity), uint32),
    protobuffs:encode(6, Message, string),
    protobuffs:encode(7, opt(Module, <<">>), string),
    protobuffs:encode(8, opt(Function, <<">>), string),
    protobuffs:encode(9, opt(Line, <<">>), string),
    protobuffs:encode(10, opt(Pid, <<">>), string)
]).
```

# JSON

```
encode_json_packet(Node) ->
  jiffy:encode([
    {packet_version, 1},
    {node, list_to_binary(Node)},
    {role, <<"calculator">>},
    {version, <<"0.0.1">>},
    {severity, 128},
    {message, <<"Calculator request 5 * 4">>},
    {module, <<"http_calculator">>},
    {function, <<"rpn">>},
    {line, <<"58">>},
    {pid, <<" ">>}
  ]}).
```

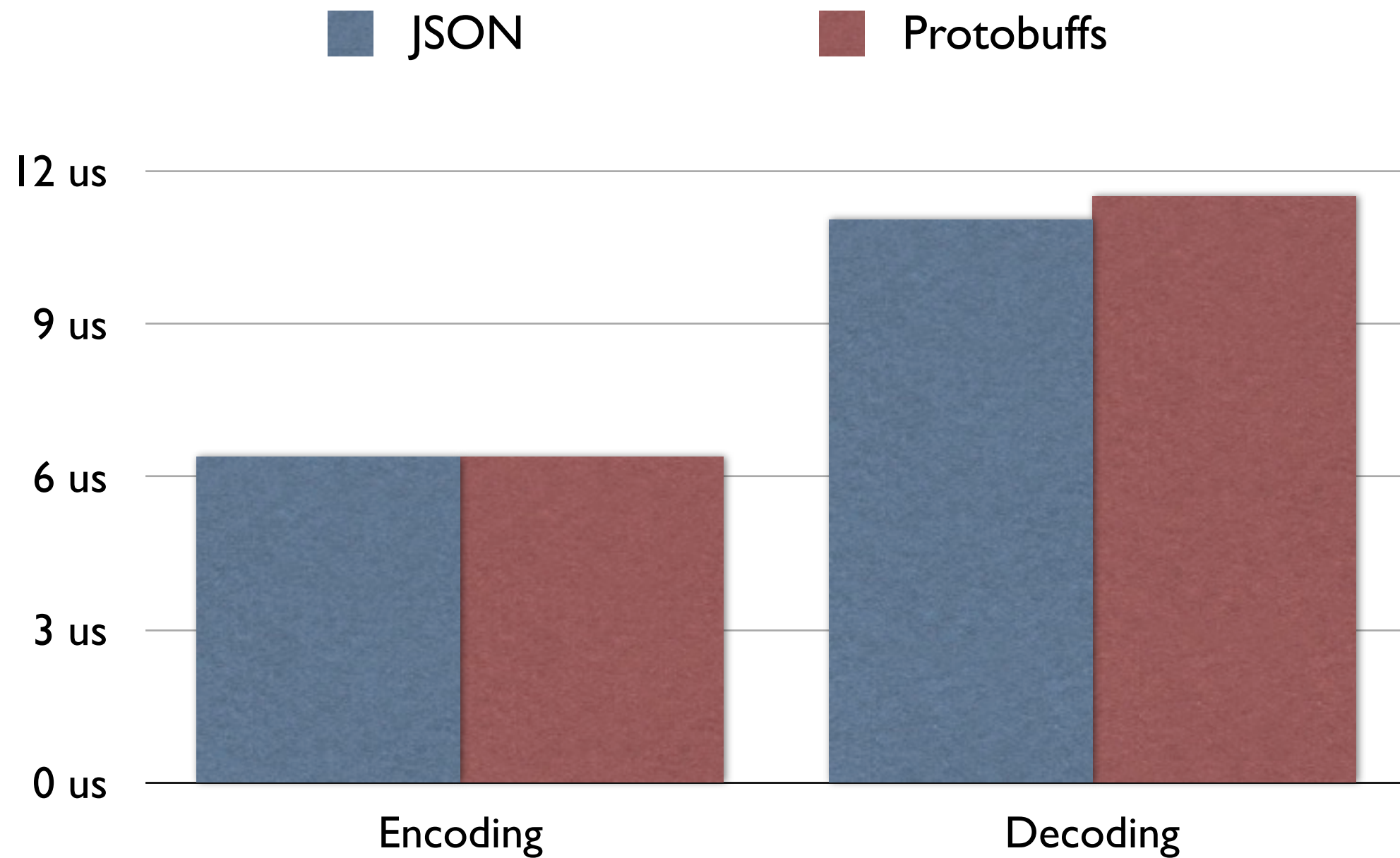
# Bits and Bytes

```
^H^A^Ropcorn@anduin.tigertext.me^Z  
calculator"^E0.0.1(^A2^XCalculator request 5 *  
4:^0http_calculatorB^CrpnJ^B58R^@
```

```
{"packet_version":1,  
"node":"popcorn@anduin.tigertext.me",  
"role":"calculator", "version":"0.0.1", "severity":128,  
"message":"Calculator request 5 * 4",  
"module":"http_calculator",  
"function":"rpn","line":"58","pid":""}
```

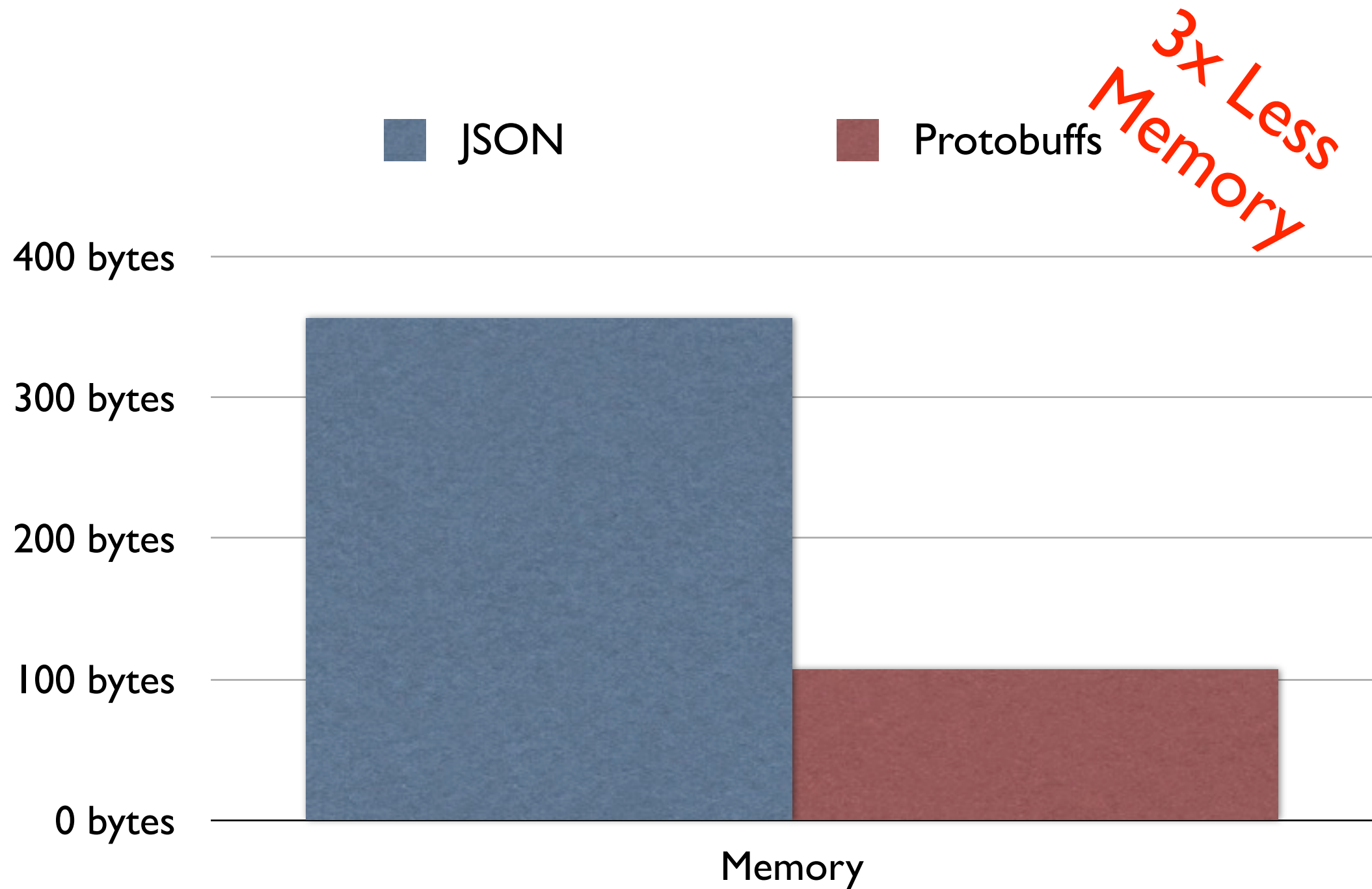
# Processing Time

Over 1,000,000 Requests on Apple Air with RI6B





# Memory



# Rendering Architecture

**Crossfilter**

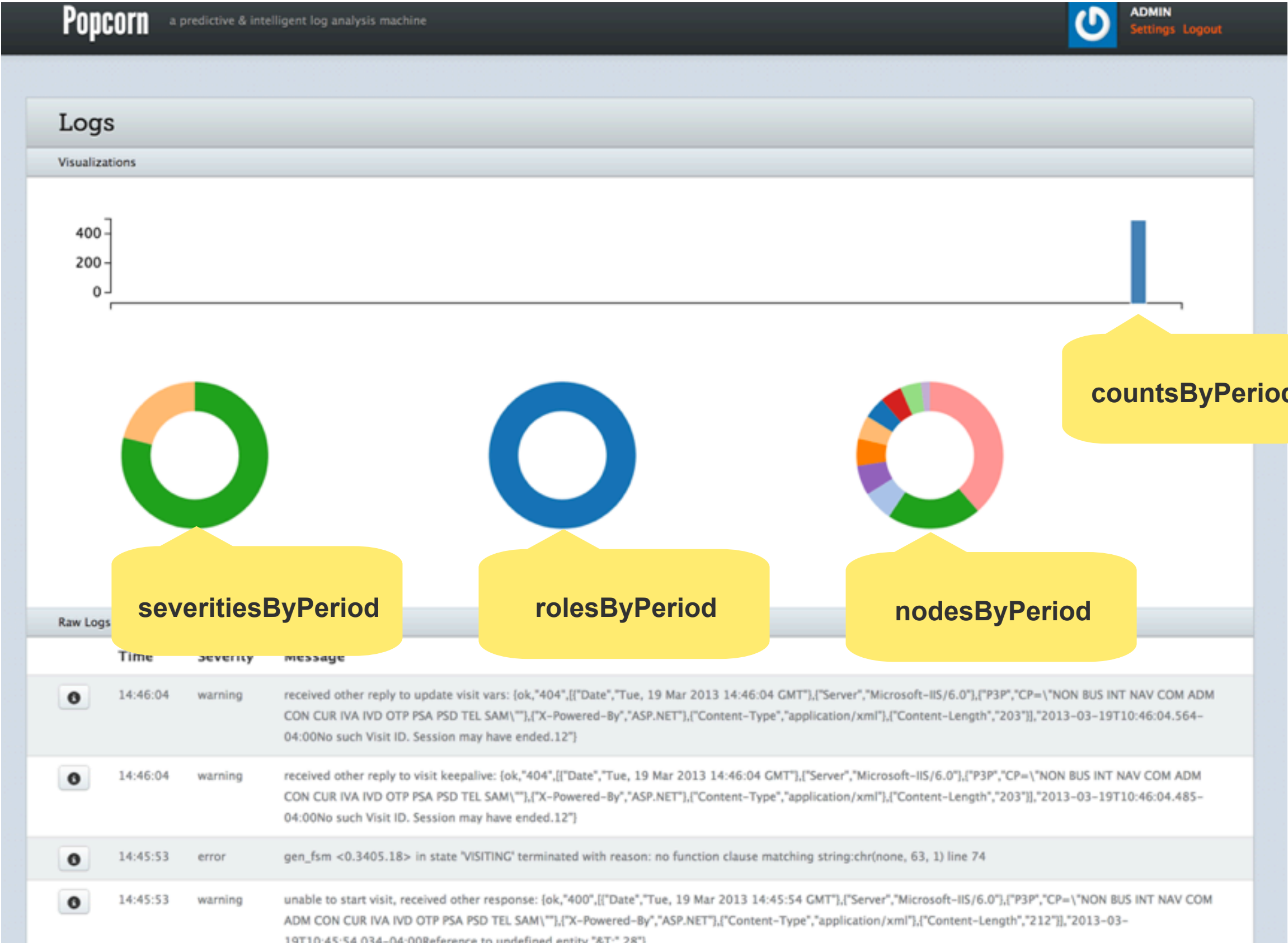
**Used only for  
grouping  
streaming data,  
destroyed every  
few seconds**

**D3**

**Rendering**

**Plain Old  
Javascript**

**Store the grouped  
data from**



# Scaling Popcorn

## Scalability Goal

Handle  $n \log$  messages per second

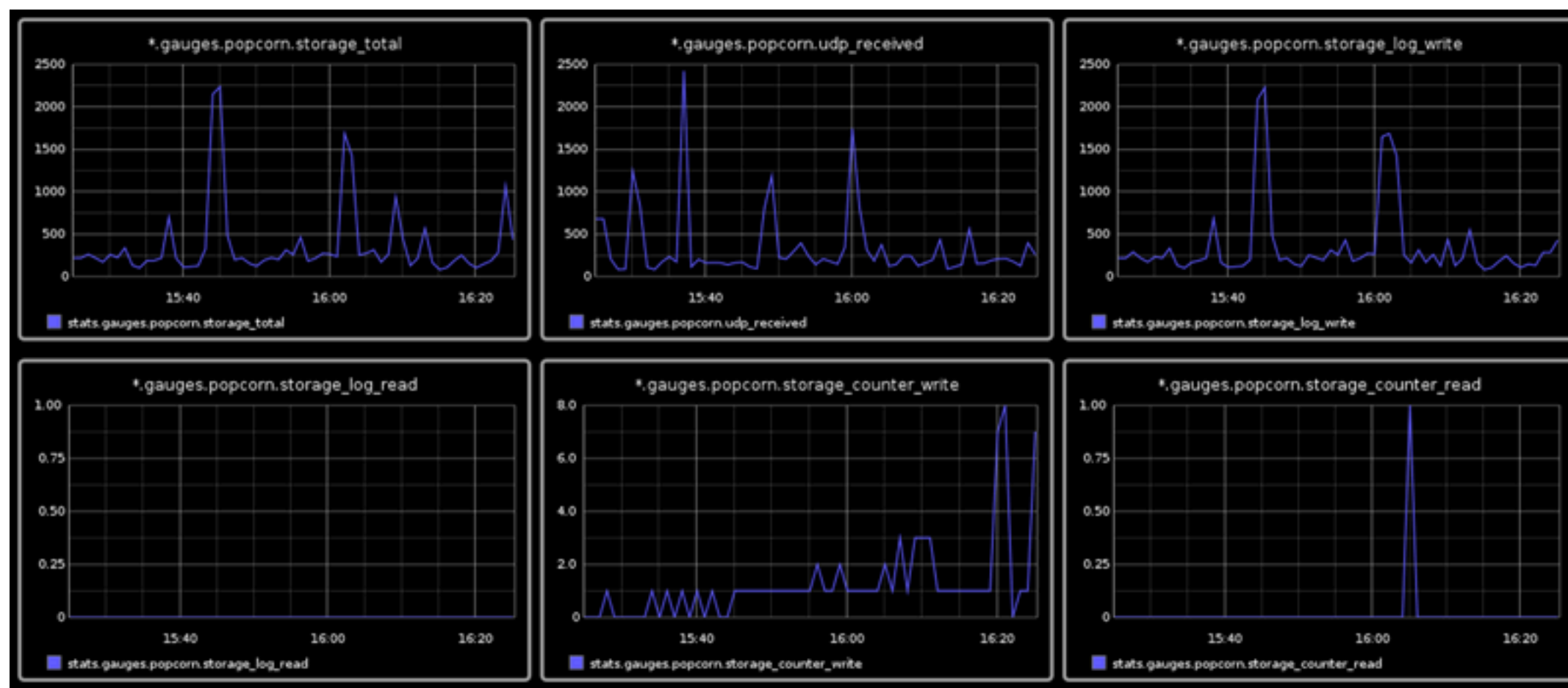
10 ✓

50 ✓

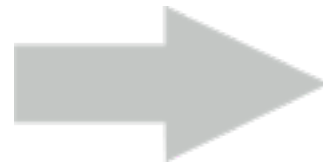
500 ✓

5000

# Let's Get Some Tools



**rps (mhald)**



**statsd &  
graphite**

**dtop**

**redbug**

**statsd**

**graphite**

**Chrome  
Activity  
Monitor**



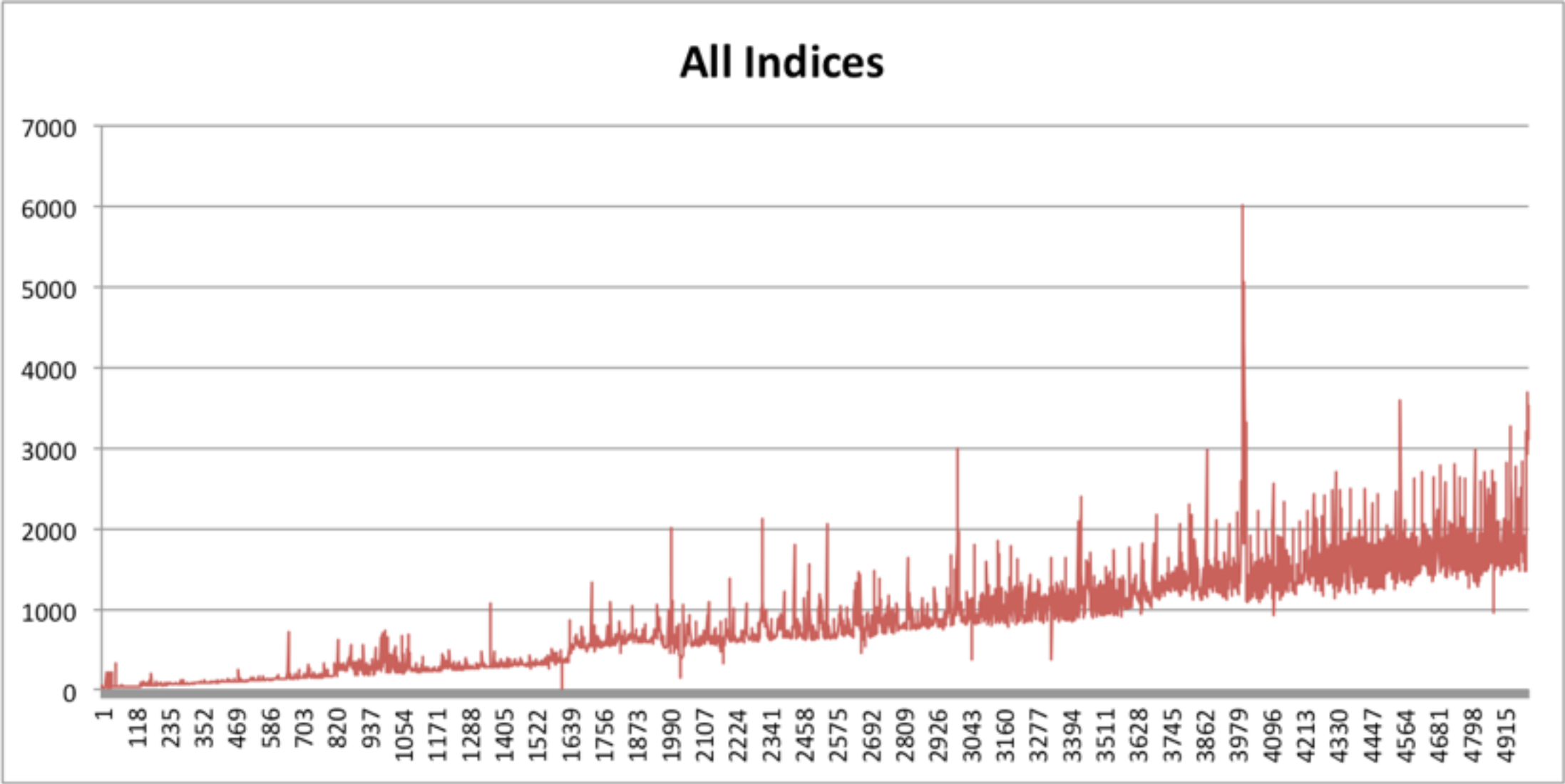
**Mnesia**

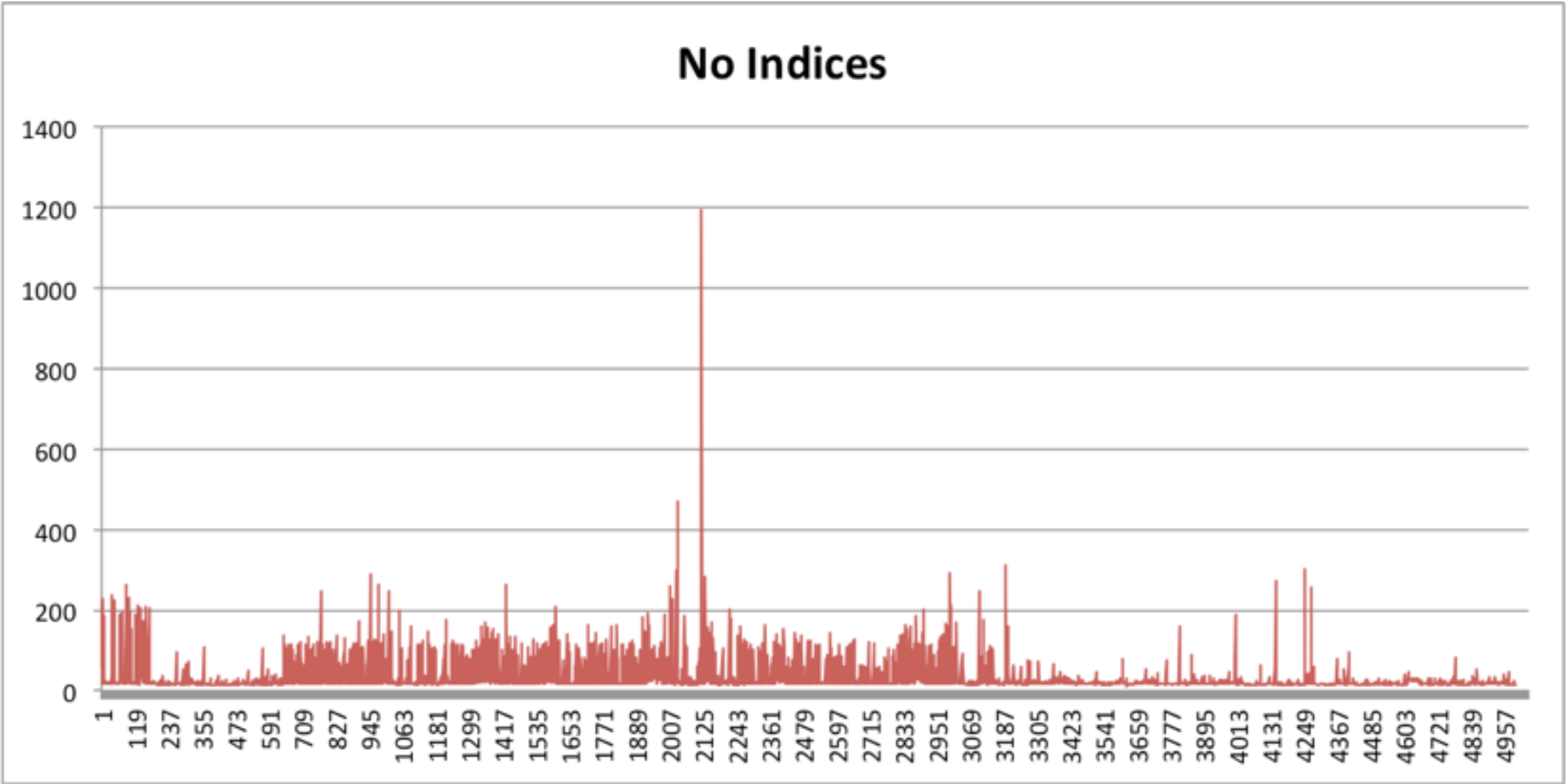
**UDP Receiving**

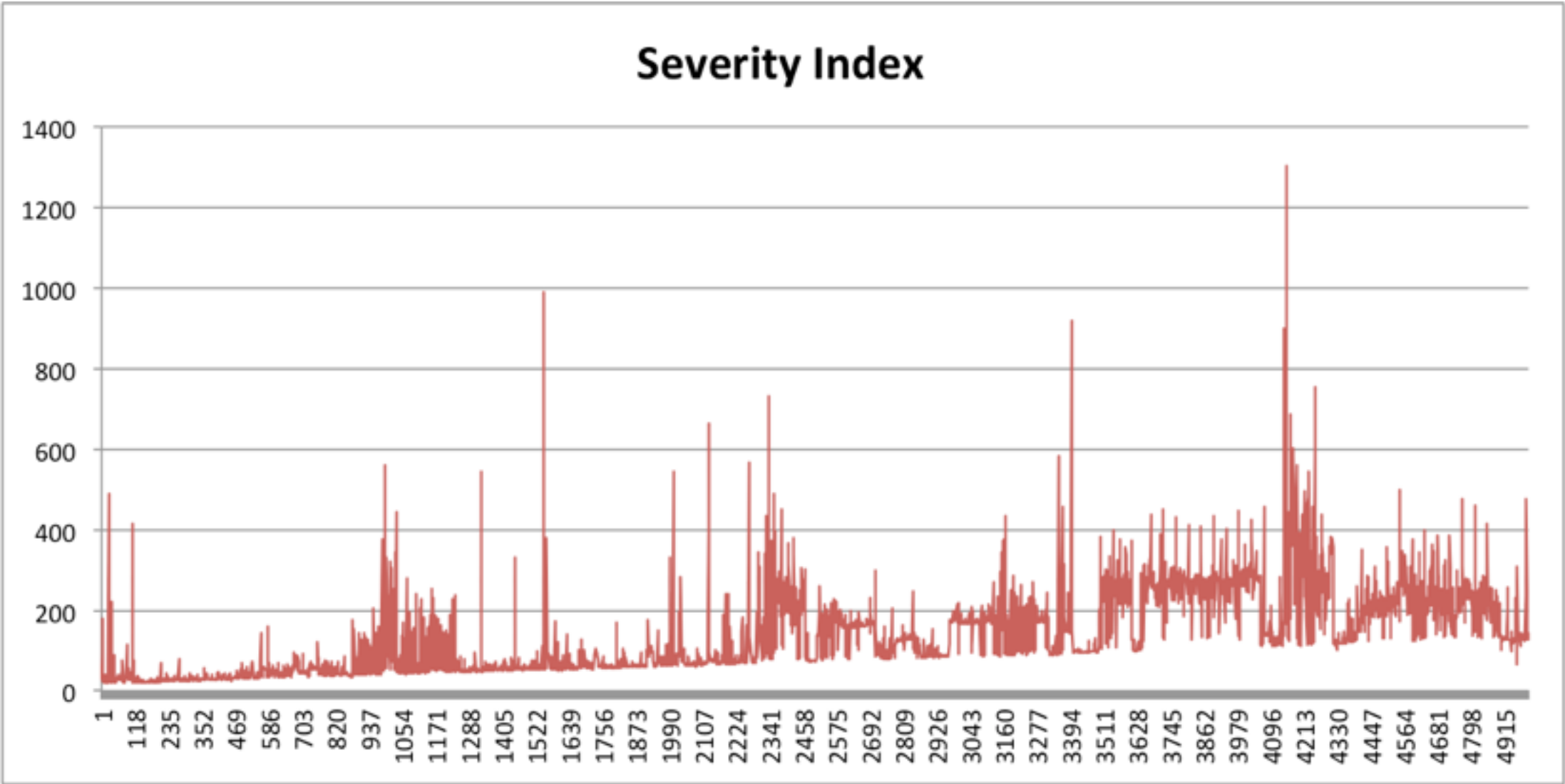
**Client (Browser)**

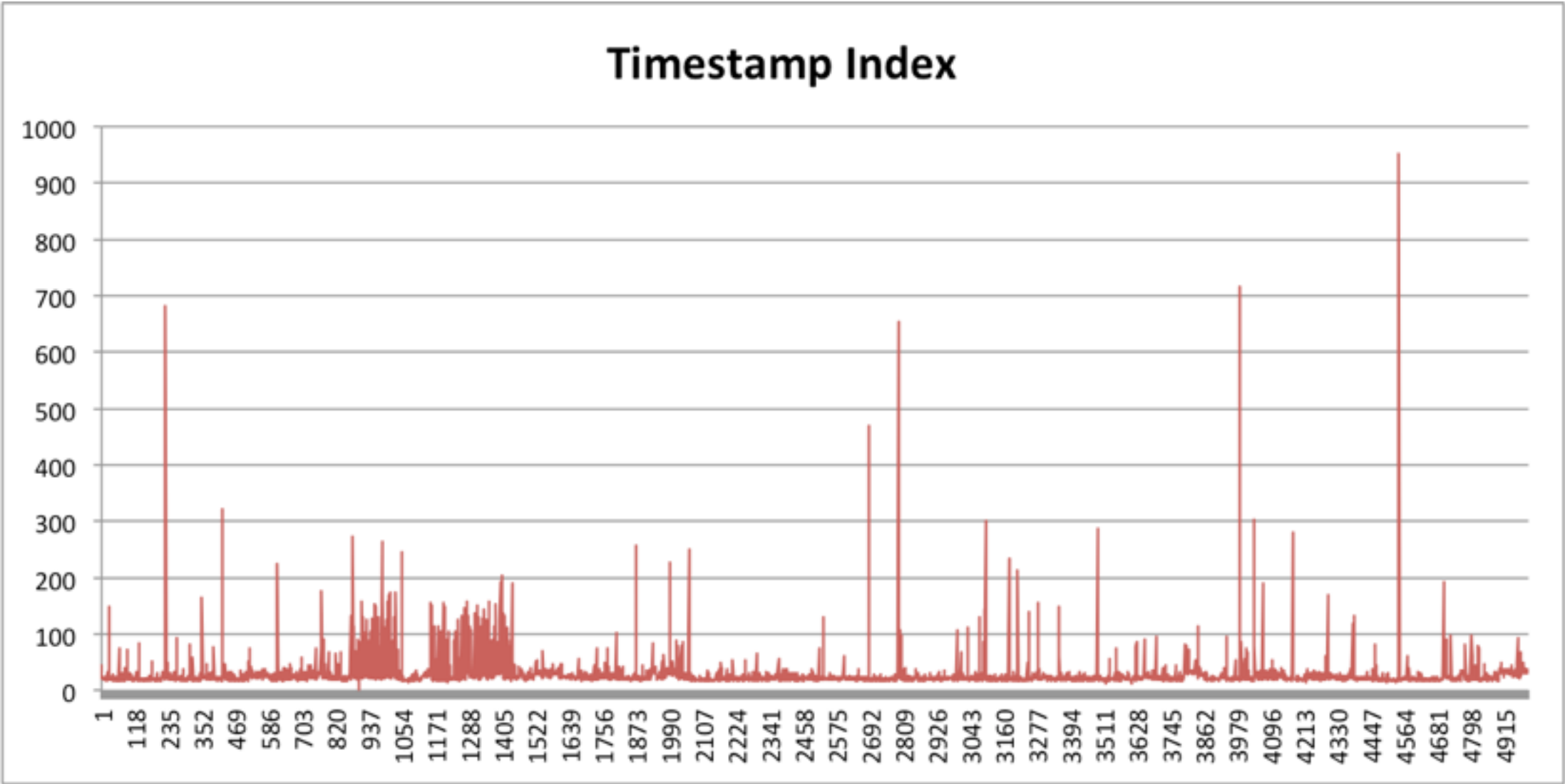
# Issue: Mnesia











# Issue: The Browser

**Executing on the server  
(Erlang)**

**vs**

**Executing in the browser  
(Javascript)**