

TEN BILLION A DAY, ONE-HUNDRED MILLISECONDS PER

**MONITORING REAL-TIME
BIDDING AT ADROLL**

I DO THINGS WITH / TO
COMPUTERS.

I CARE ABOUT RELIABLE,
COMPLEX AND CRITICAL SYSTEMS.

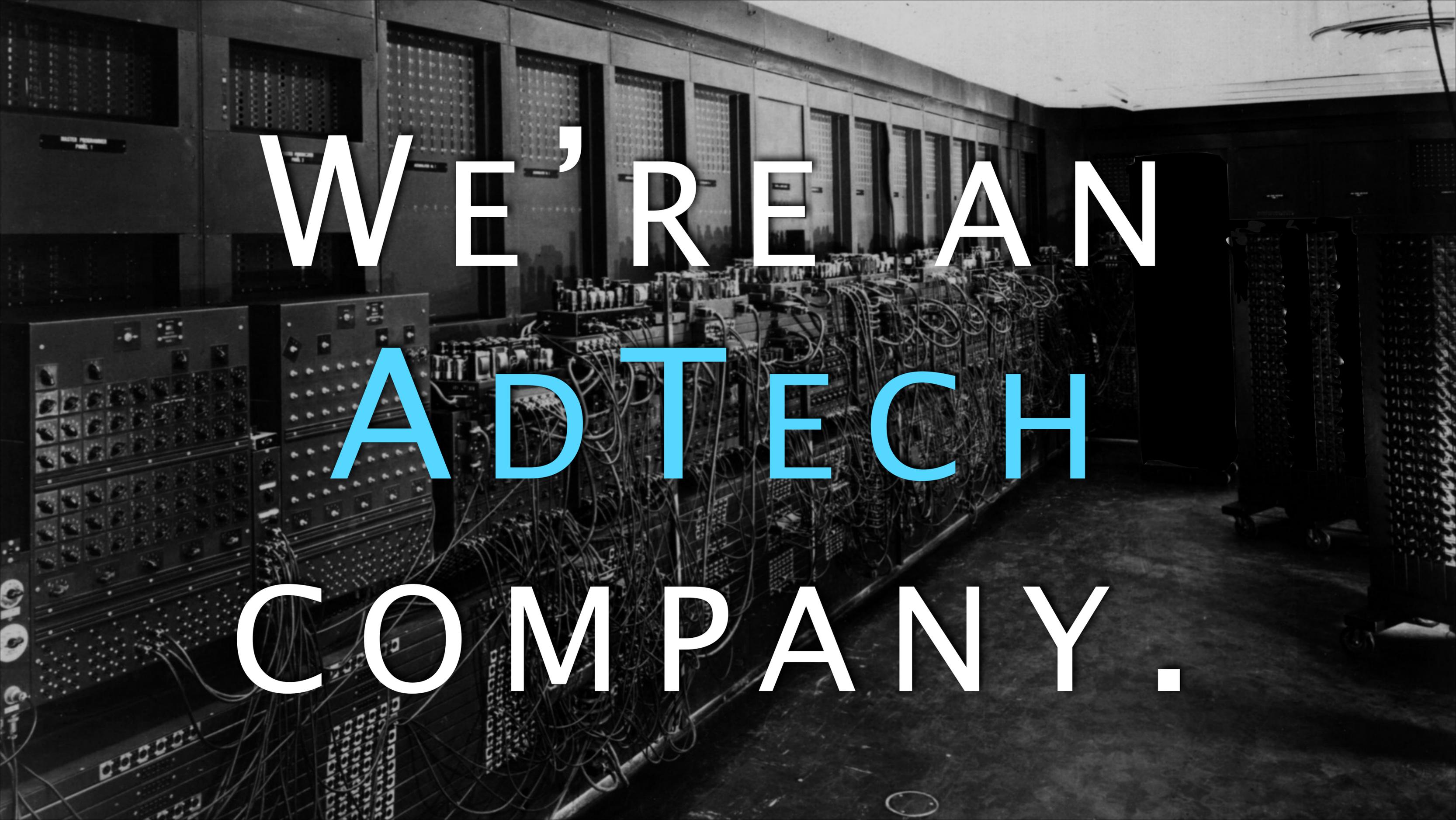
ADROLL



LESS THAN THIS

MORE THIS





WE'RE AN
ADTECH
COMPANY.

RETARGETING, IT'S
ALL ABOUT DATA.

- Our customers want to show ads to people who might care to see them.
- We partner with “exchanges” to enter ad-slot auctions.
- These auctions are executed and finalized while consumer’s webpages load.



REAL -

TIME

BIDDING

the nature of the problem domain:

- Low latency (< 100ms per transaction)
- Firm real-time system
- Highly concurrent (> 30 billion transactions per day)
- Global, 24/7 operation

"HUMANS ARE BAD AT PREDICTING THE PERFORMANCE OF COMPLEX SYSTEMS(...). OUR ABILITY TO CREATE LARGE AND COMPLEX SYSTEMS FOOLS US INTO BELIEVING THAT WE'RE ALSO ENTITLED TO UNDERSTAND THEM."

-CARLOS BUENO

"MATURE OPTIMIZATION HANDBOOK"

AHEAD OF TIME
VERIFICATION IS
NOT SUFFICIENT.

(DON'T SCRIMP ON IT, THOUGH.)

IGNORANCE AND
COMPLEX INTERACTIONS
WITH EXTERNAL
SYSTEMS ARE WHY WE
CAN'T HAVE NICE
THINGS.



AT SCALE, EVEN RARE
EVENTS HAPPEN
FREQUENTLY.

AT SCALE, BAD
THINGS HAPPEN
FASTER THAN
HUMANS CAN
RESPOND.





THE RESULTS ARE
RARELY PRETTY.

THE CAUSES ARE RARELY
QUICK TO DISCOVER.





WHAT
CAN BE
DONE?

WE HAVE TO OBSERVE
OUR SYSTEMS WHILE
THEY RUN.

NOT ALL SYSTEMS ARE
MONITORABLE, JUST AS NOT
ALL ARE TESTABLE.

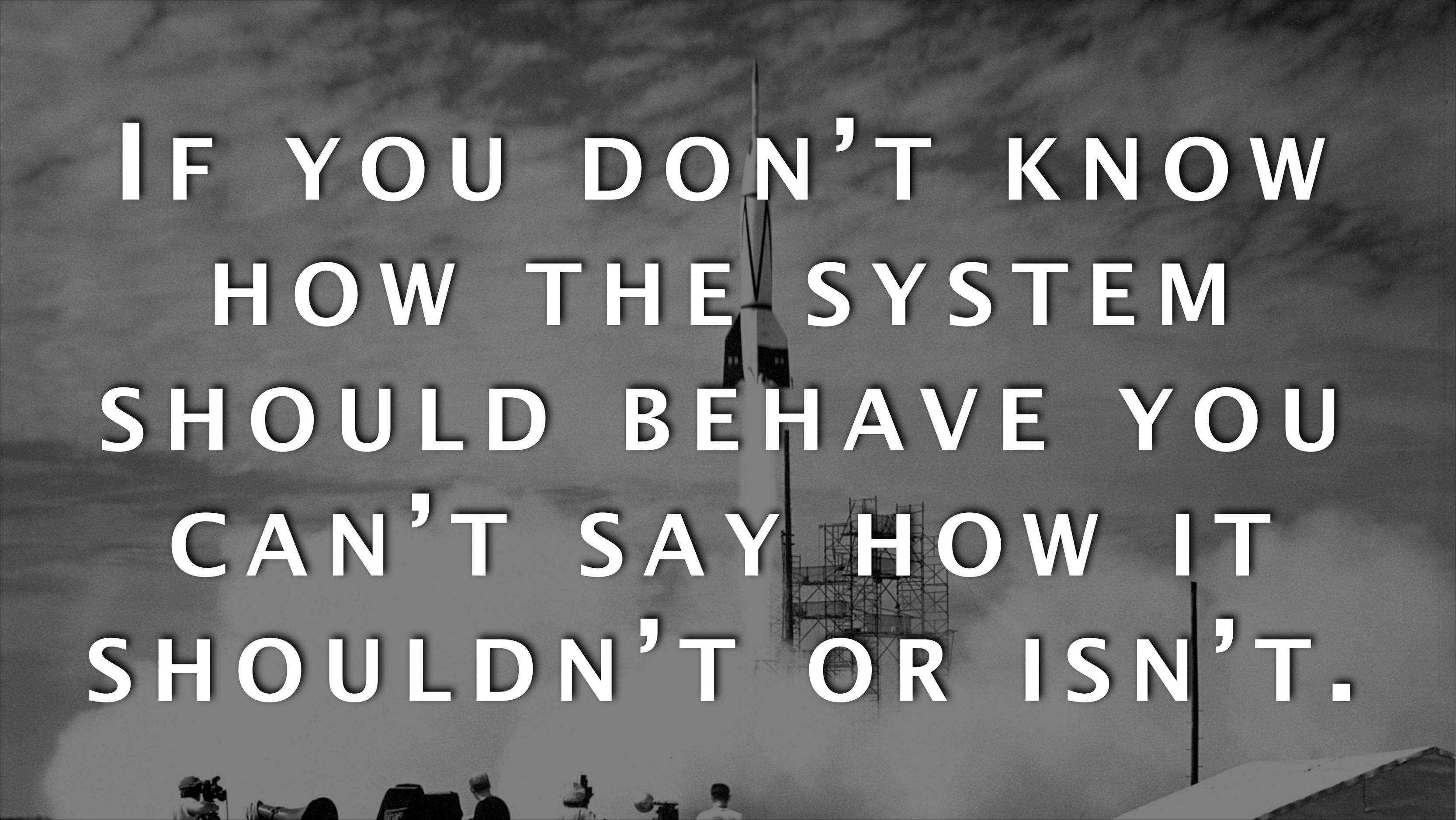
IT'S A MATTER OF DESIGN
AND OF CULTURE.

A DESIGN THAT IS MONITORABLE
TAKES STOCK OF ITS INTERNAL
TOLERANCES, EXTERNAL
INTERFACES AND EXPOSES THESE
FOR AN OPERATOR.

AN **ENGINEER** IS RESPONSIBLE FOR
DESIGNING AND BUILDING
SYSTEMS.

AN **OPERATOR** IS RESPONSIBLE
FOR RUNNING THEM.

AHEAD-OF-TIME
VERIFICATION—TESTING,
TYPE-CHECKING—GIVES THE
ENGINEER INTUITION ABOUT
THE SYSTEM.



IF YOU DON'T KNOW
HOW THE SYSTEM
SHOULD BEHAVE YOU
CAN'T SAY HOW IT
SHOULDN'T OR ISN'T.

MONITORING GIVES THE
OPERATOR INFORMATION
ABOUT THE BEHAVIOR OF
THE RUNNING SYSTEM.

THE OPERATORS AND ENGINEERS
ARE OFTEN THE SAME PEOPLE.

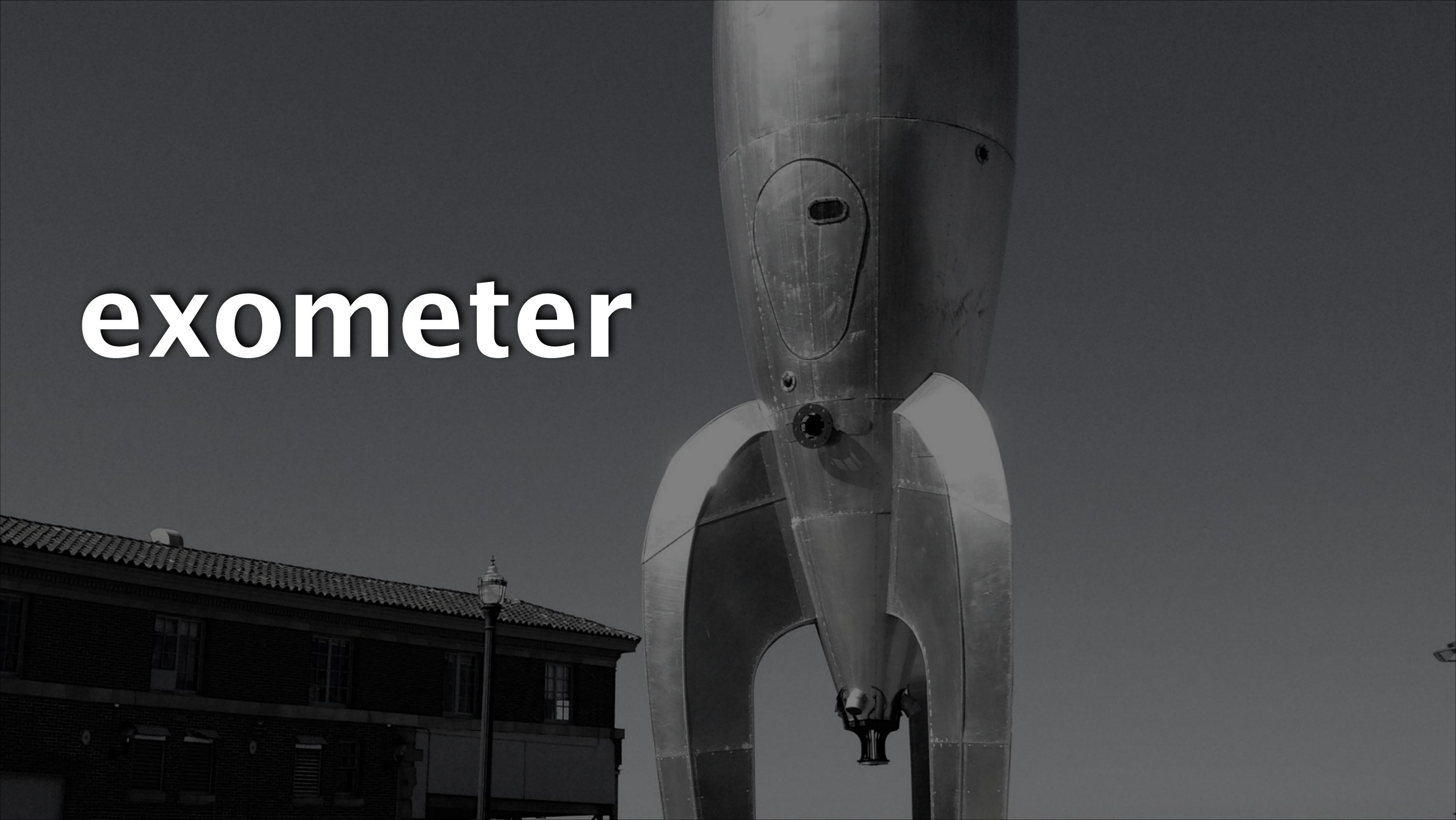
THERE'S A POTENTIAL FOR A POSITIVE
FEEDBACK LOOP OF QUALITY HERE.

"WHILE THE SKILL OF REENTRY WAS EASILY HANDLED BY AUTOMATED SYSTEMS, THE PILOT'S PRIMARY FUNCTION EVOLVED TO BE A REDUNDANT SYSTEM (...) COORDINATING A VARIETY OF CONTROLS AS MUCH AS DIRECTLY CONTROLLING THE VEHICLE."

-DAVID A. MINDELL

DIGITAL APOLLO: HUMAN AND MACHINE IN SPACEFLIGHT

exometer



`exometer` — github.com/Feuerlabs/exometer

- Created by Feuerlabs.
- Responsive upstream (Ulf Wiger never sleeps?)
- Metric collection, aggregation and reporting decoupled.
- Static and dynamic configuration.
- Very low, predictable runtime overhead.

IMPORTANT TERMS

- **METRIC:** a measurement
- **ENTRY:** a receiver and aggregator of metrics
- **REPORTER:** an entity which samples entries on a regular interval and optionally ships these samples onto a third-system
- **SUBSCRIPTION:** the definition of the regular interval on which reporters sample entries

Defining Entries

```
{predefined, [
  {erlang, memory},
  {function, erlang, memory,
   ['$dp'], value, [ets, binary]},
  []
}],

{erlang, statistics},
{function, erlang, statistics,
 ['$dp'], value, [run_queue]},
[]
},

{erlang, gc},
{function, erlang, statistics,
  [garbage_collection], match,
  {total_coll, rec_wrd, '_'}},
[]
},

{boodah, freq_cap, not_found},
spiral},
{boodah, freq_cap, ok}, spiral},
{boodah, freq_cap, timeout}, spiral}
]],
```

Defining Entries

```
{predefined, [
  {erlang, memory},
  {function, erlang, memory,
   ['$dp'], value, [ets, binary]},
  []
},

{erlang, statistics},
{function, erlang, statistics,
 ['$dp'], value, [run_queue]},
[]
},

{erlang, gc},
{function, erlang, statistics,
  [garbage_collection], match,
  {total_coll, rec_wrd, '_'}},
[]
},

{boodah, freq_cap, not_found},
spiral},
{boodah, freq_cap, ok}, spiral},
{boodah, freq_cap, timeout}, spiral}
]],
```

Defining Entries

```
{predefined, [  
  {erlang, memory},  
  {function, erlang, memory,  
    ['$dp'], value, [ets, binary]},  
  []  
},  
  
{erlang, statistics},  
{function, erlang, statistics,  
  ['$dp'], value, [run_queue]},  
  []  
},
```

```
{erlang, gc},  
  {function, erlang, statistics,
```

```
    [garbage_collection], match,  
    {total_coll, rec_wrd, '_'}},  
  []  
},  
  
  {boodah, freq_cap, not_found},  
  spiral},  
  {boodah, freq_cap, ok}, spiral},  
  {boodah, freq_cap, timeout}, spiral}  
  ]},
```

Defining Entries

```
{predefined, [
  {erlang, memory},
  {function, erlang, memory,
   ['$dp'], value, [ets, binary]},
  []
},

{erlang, statistics},
{function, erlang, statistics,
 ['$dp'], value, [run_queue]},
[]
},

{erlang, gc},
{function, erlang, statistics,
  [garbage_collection], match,
  {total_coll, rec_wrd, '_'}},
[]
},

{boodah, freq_cap, not_found},
spiral},
{boodah, freq_cap, ok}, spiral},
{boodah, freq_cap, timeout}, spiral}
]],
```

Defining Entries

```
{predefined, [
  {erlang, memory},
  {function, erlang, memory,
   ['$dp'], value, [ets, binary]},
  []
},

{erlang, statistics},
{function, erlang, statistics,
 ['$dp'], value, [run_queue]},
[]
},

{erlang, gc},
{function, erlang, statistics,
  [garbage_collection], match,
  {total_coll, rec_wrd, '_'}},
[]
},

{boodah, freq_cap, not_found},
spiral},
{boodah, freq_cap, ok}, spiral},
{boodah, freq_cap, timeout}, spiral}
]],
```

Defining Reporters

```
{ reporters,  
  [  
    { exometer_report_statsd,  
      [  
        {hostname, "localhost"},  
        {port, 8125},  
        {type_map,  
          [  
            {erlang,statistics,run_queue},  
            histogram},  
            {erlang,gc,tot_coll},  
            histogram},  
            {erlang,gc,rec_wrd},  
            histogram},  
            {erlang,memory,ets}, gauge},  
            {erlang,memory,binary},gauge},  
            {boodah,freq_cap,not_found},gauge  
          ],  
          {boodah,freq_cap,ok},gauge},  
          {boodah,freq_cap,timeout},gauge}  
        ]},  
    ]},  
  ]},  
}
```

Defining Reporters

```
{ reporters,  
  [  
    { exometer_report_statsd,  
      [  
        {hostname, "localhost"},  
        {port, 8125},  
        {type_map,  
          [  
            {erlang,statistics,run_queue},  
            histogram},  
            {erlang,gc,tot_coll},  
            histogram},  
            {erlang,gc,rec_wrd},  
            histogram},  
            {erlang,memory,ets}, gauge},  
            {erlang,memory,binary},gauge},  
            {boodah,freq_cap,not_found},gauge  
          ],  
          {boodah,freq_cap,ok},gauge},  
          {boodah,freq_cap,timeout},gauge}  
        ]},  
    ]},  
  ]},  
}
```

Defining Reporters

```
{ reporters,  
  [  
    { exometer_report_statsd,  
      [  
        {hostname, "localhost"},  
        {port, 8125},  
        {type_map,  
          [  
            {[erlang,statistics,run_queue],  
             histogram},  
            {[erlang,gc,tot_coll],  
             histogram},  
            {[erlang,gc,rec_wrd],  
             histogram},  
            {[erlang,memory,ets], gauge},  
            {[erlang,memory,binary],gauge},  
            {[boodah,freq_cap,not_found],gauge  
            },  
            {[boodah,freq_cap,ok],gauge},  
            {[boodah,freq_cap,timeout],gauge}  
          ]}],  
      }  
    ]  
  }  
}
```

Defining Subscriptions

```
{ report,  
  [  
    { subscribers,  
      [  
        { exometer_report_statsd, [erlang, statistics],  
          run_queue, 1000},  
  
        { exometer_report_statsd, [erlang, gc],  
          tot_coll, 1000},  
        { exometer_report_statsd, [erlang, gc],  
          rec_wrd, 1000},  
  
        { exometer_report_statsd, [erlang, memory],  
          ets, 10000},  
        { exometer_report_statsd, [erlang, memory],  
          binary, 10000},  
  
        { exometer_report_statsd,  
          [boodah, freq_cap, not_found], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, ok], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, timeout], one, 1000}  
      ]  
    }  
  ]  
}
```

Defining Subscriptions

```
{ report,  
  [  
    { subscribers,  
      [  
        { exometer_report_statsd, [erlang, statistics],  
          run_queue, 1000},  
  
        { exometer_report_statsd, [erlang, gc],  
          tot_coll, 1000},  
        { exometer_report_statsd, [erlang, gc],  
          rec_wrd, 1000},  
  
        { exometer_report_statsd, [erlang, memory],  
          ets, 10000},  
        { exometer_report_statsd, [erlang, memory],  
          binary, 10000},  
  
        { exometer_report_statsd,  
          [boodah, freq_cap, not_found], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, ok], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, timeout], one, 1000}  
      ]  
    }  
  ]  
}
```

Defining Subscriptions

```
{ report,  
  [  
    { subscribers,  
      [  
        { exometer_report_statsd, [erlang, statistics],  
          run_queue, 1000},  
  
        { exometer_report_statsd, [erlang, gc],  
          tot_coll, 1000},  
        { exometer_report_statsd, [erlang, gc],  
          rec_wrd, 1000},  
  
        { exometer_report_statsd, [erlang, memory],  
          ets, 10000},  
        { exometer_report_statsd, [erlang, memory],  
          binary, 10000},  
  
        { exometer_report_statsd,  
          [boodah, freq_cap, not_found], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, ok], one, 1000},  
        { exometer_report_statsd,  
          [boodah, freq_cap, timeout], one, 1000}  
      ]  
    }  
  ]  
}
```

Doing it dynamically.

```
1> exometer:new([a, histogram], histogram).  
ok
```

```
2> exometer:get_value([a, histogram]).  
{ok, [{n, 0},  
      {mean, 0},  
      {min, 0},  
      {max, 0},  
      {median, 0},  
      {50, 0},  
      {75, 0},  
      {90, 0},  
      {95, 0},  
      {99, 0},  
      {999, 0}]}
```

```
3> exometer_report:add_reporter(  
    exometer_report_tty, []).  
ok
```

```
4> exometer_report:subscribe(  
    exometer_report_tty,  
    [a, histogram], mean, 1000, []).  
ok
```

```
exometer_report_tty: a_histogram_mean  
1393627070:0  
exometer_report_tty: a_histogram_mean  
1393627071:0  
exometer_report_tty: a_histogram_mean  
1393627072:0
```

WHAT ARE WE LOOKING
FOR?



- VM killers
- System performance regressions
- Abnormal system behavior
- Surprises

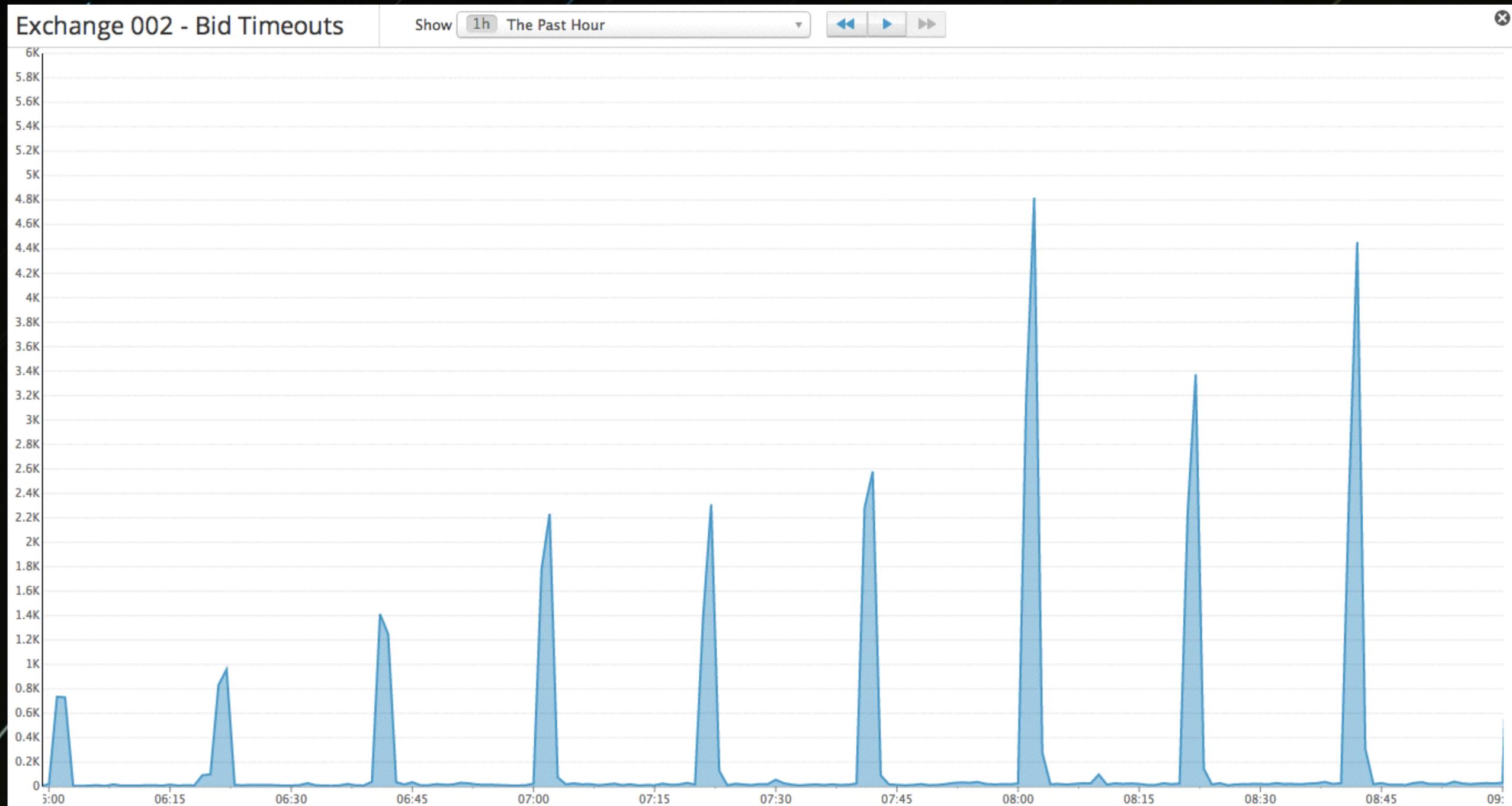
“ABNORMAL SYSTEM BEHAVIOR”?

A black and white photograph of a building's exterior. In the foreground, a dark, horizontal balcony railing is visible. The background shows a building with a curved facade and a window with a grid pattern. The text "CASE STUDIES" is overlaid in the center in a white, serif font.

CASE STUDIES

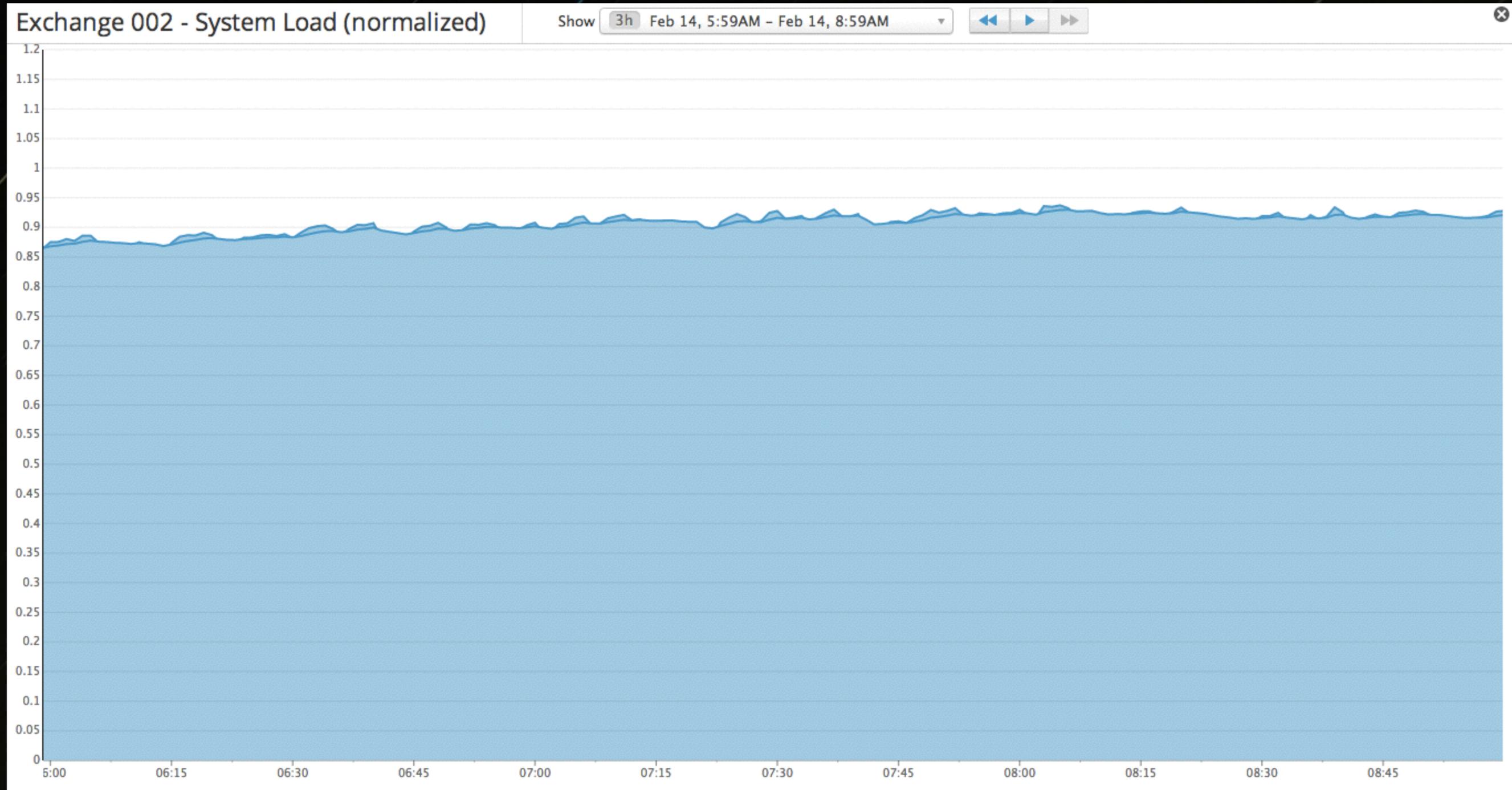
CASE STUDY:
ADROLL RTB TIMEOUT SPIKES

Case Study: AdRoll RTB Timeout Spikes



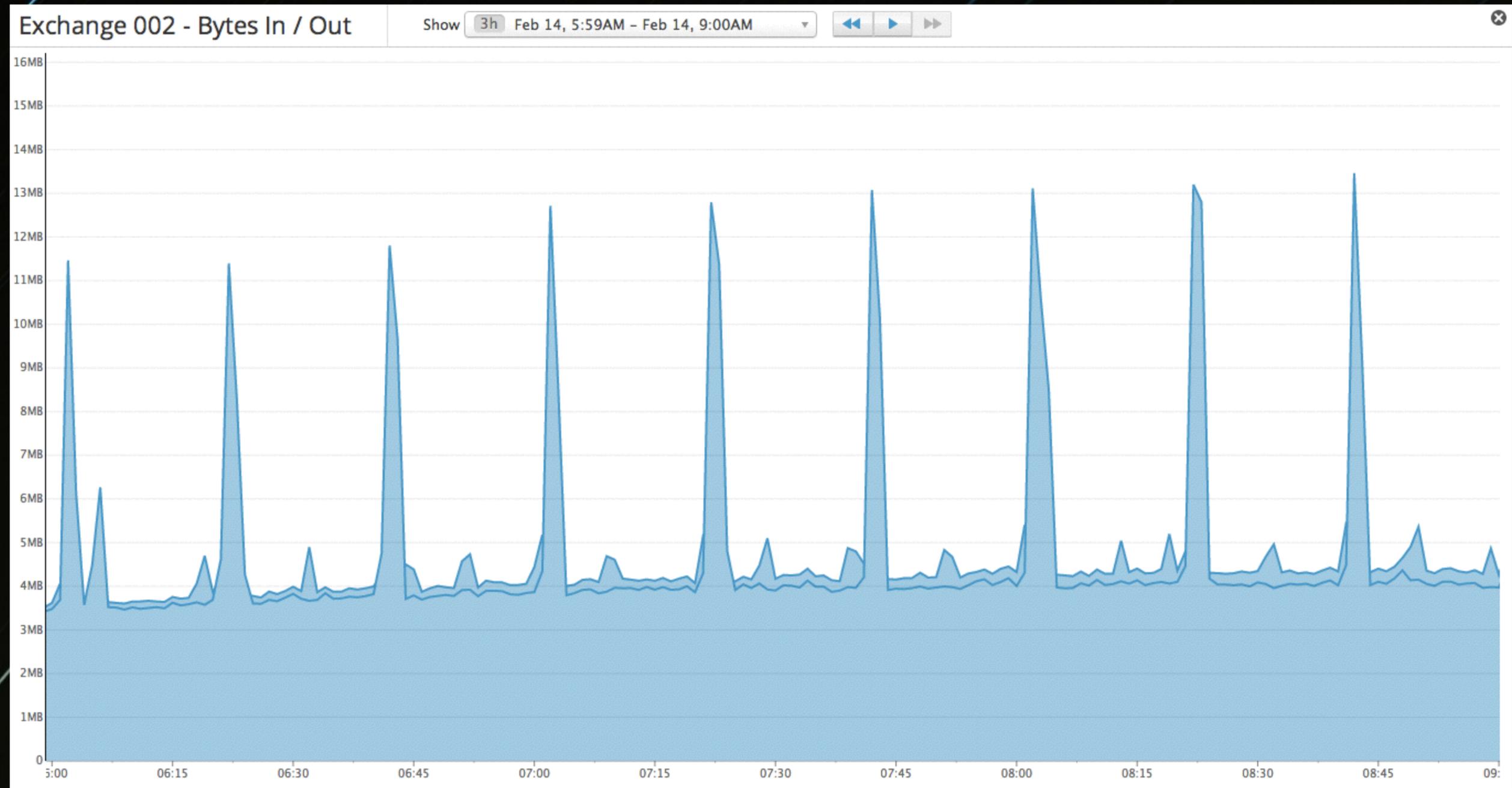
PERIODIC BID TIMEOUTS

Case Study: AdRoll RTB Timeout Spikes



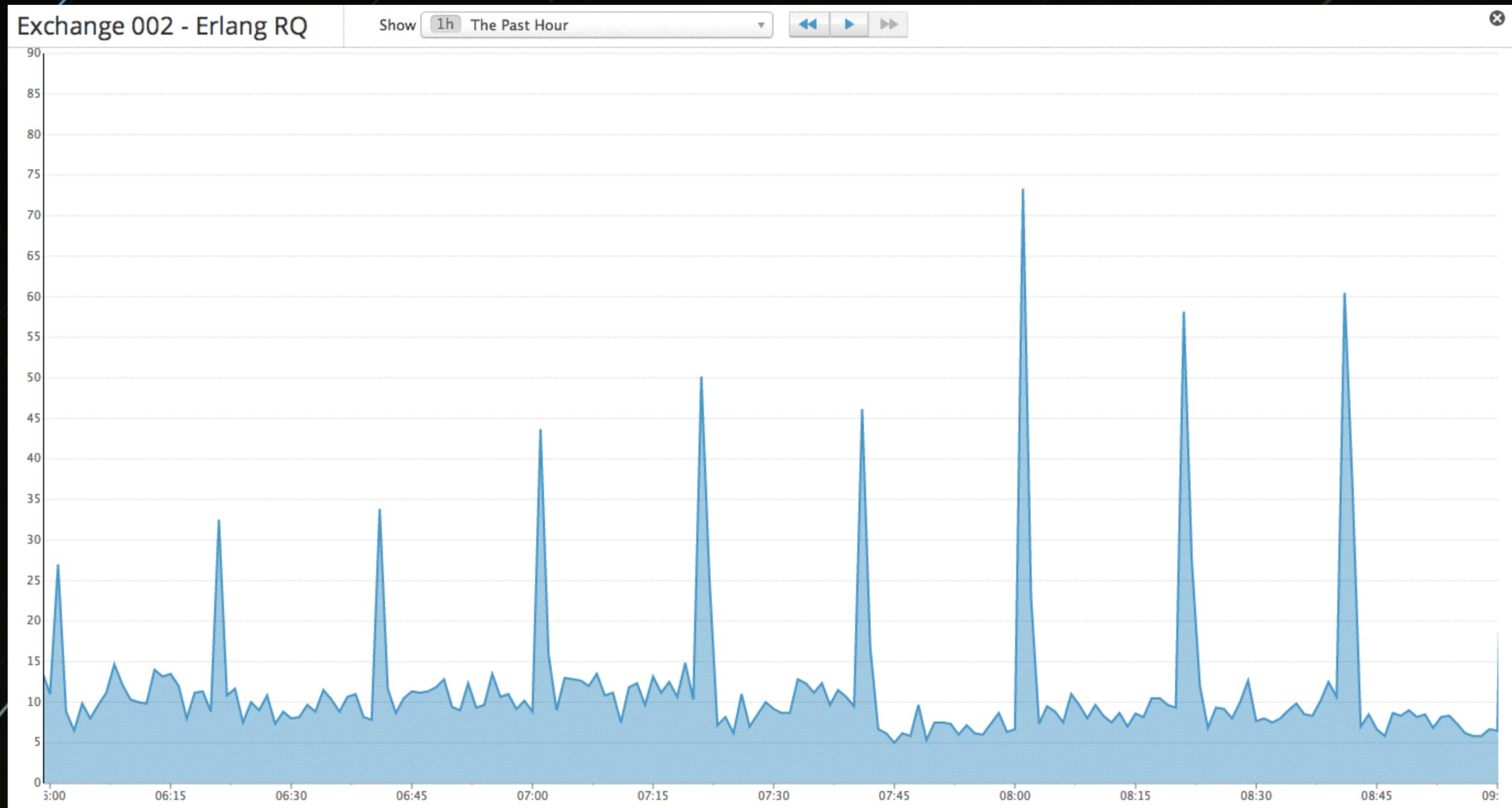
CONSISTENT SYSTEM LOAD

Case Study: AdRoll RTB Timeout Spikes



CORRELATED NETWORK TRAFFIC SPIKES

Case Study: AdRoll RTB Timeout Spikes



CORRELATED RUN QUEUE SPIKES

Case Study: AdRoll RTB Timeout Spikes

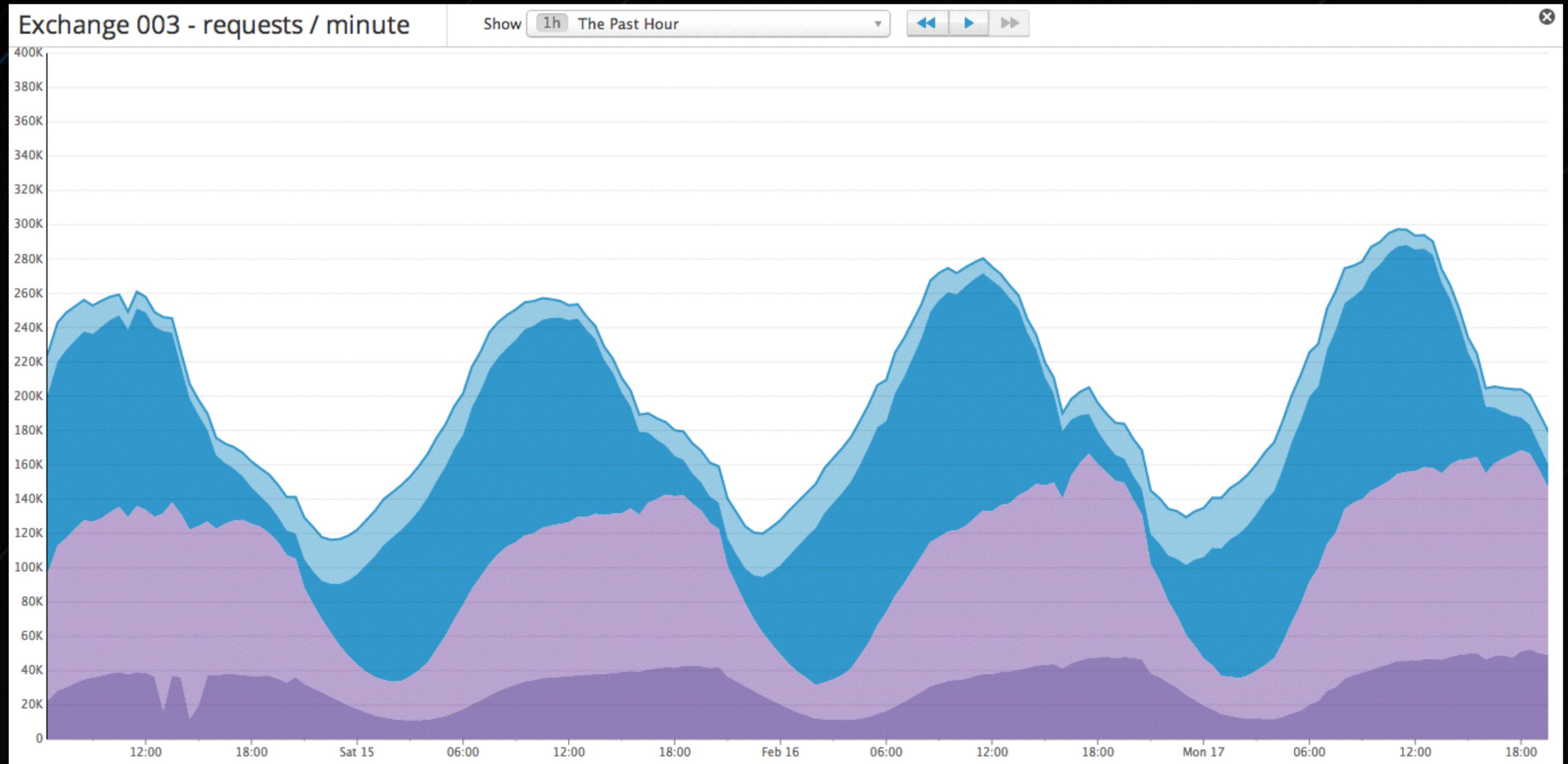
WHAT HAPPENED?

Case Study: AdRoll RTB Timeout Spikes

- Scheduler threads were locked to CPUs
- Background process comes on every 20 minutes, consumes a lot of CPU time
- No cpu-shield was set up on our production systems
- OS bumped a scheduler thread off its CPU, backing up its run-queue

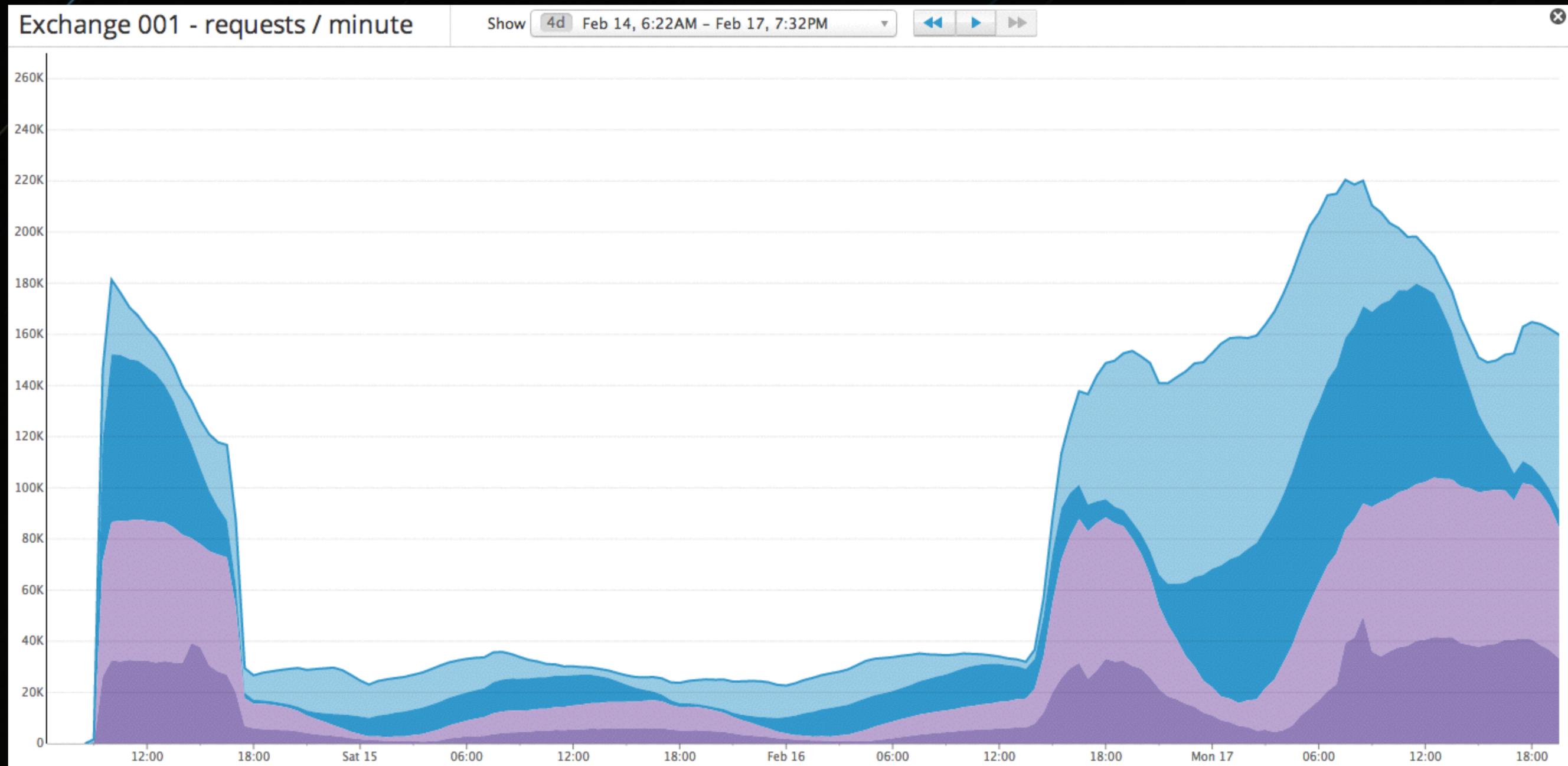
CASE STUDY: EXCHANGE THROTTLING

Case Study: Exchange Throttling



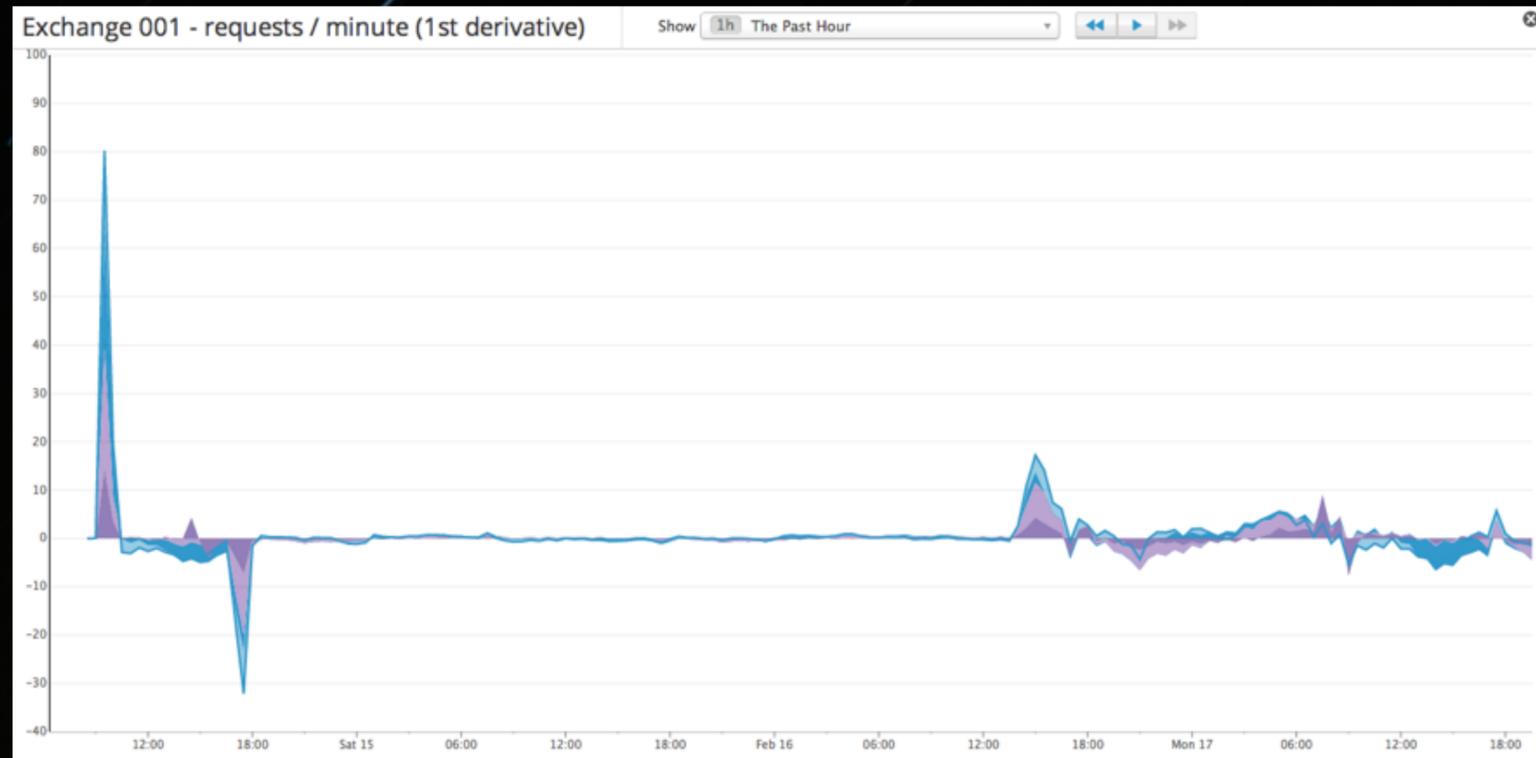
HEALTHY PATTERN OF BID REQUESTS

Case Study: Exchange Throttling

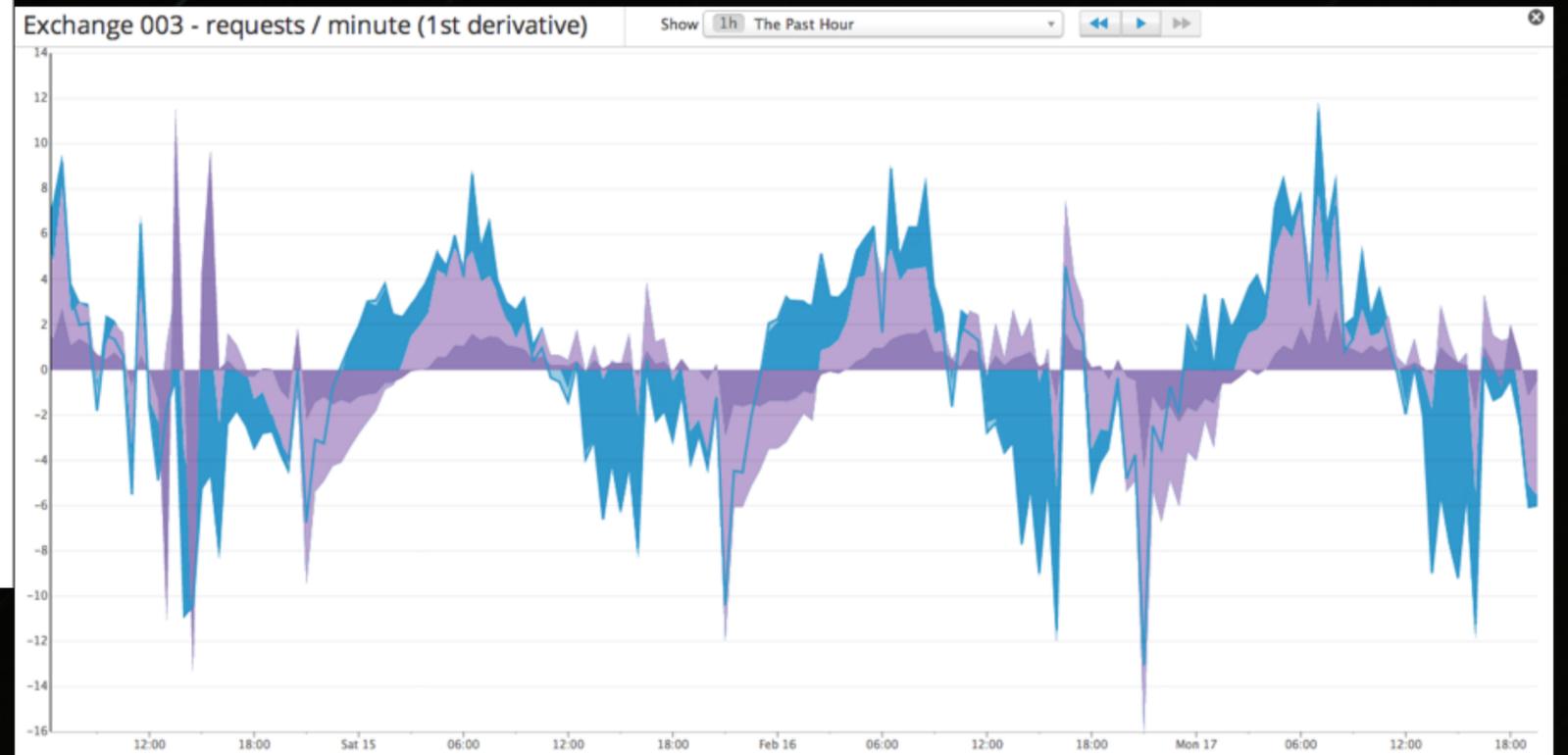


THE TROUGH OF THROTTLING

Case Study: Exchange Throttling

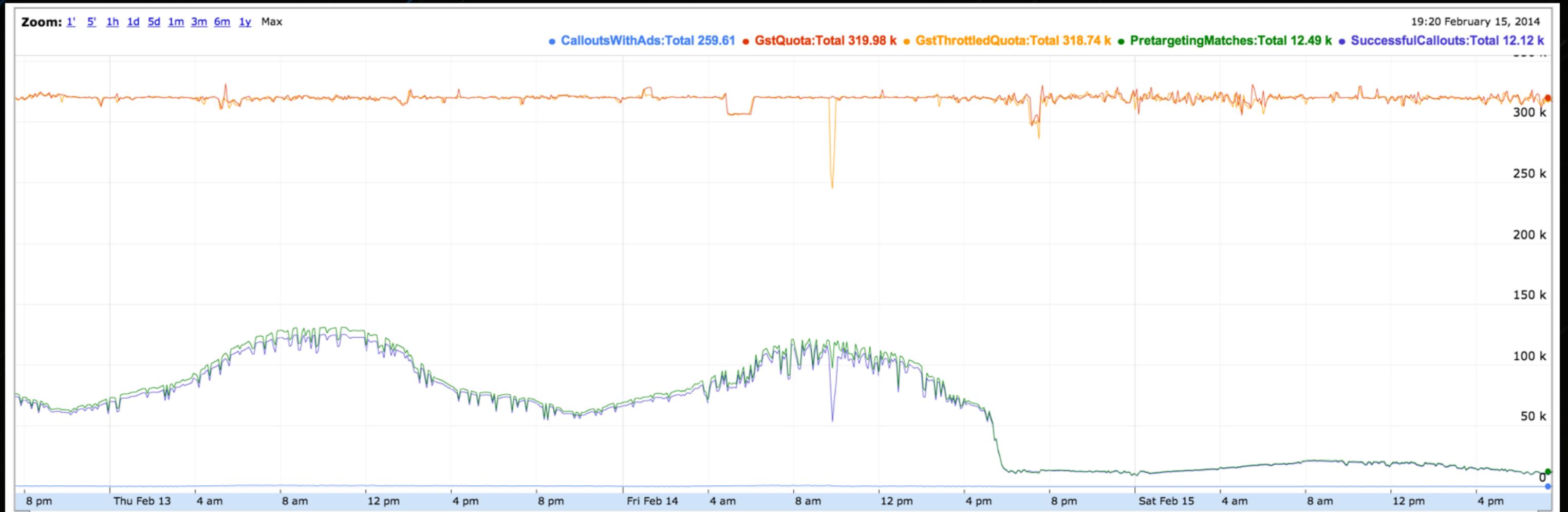


BAD



GOOD

Case Study: Exchange Throttling



PROBLEM CONFIRMED WITH EXCHANGE

Case Study: Exchange Throttling

- All other metrics (run-queue, CPU, network IO) were fine.
- Confirmed that no changes had been made to the running systems via deployment.
- Amazon data showed no network issues to our machines.

Case Study: Exchange Throttling

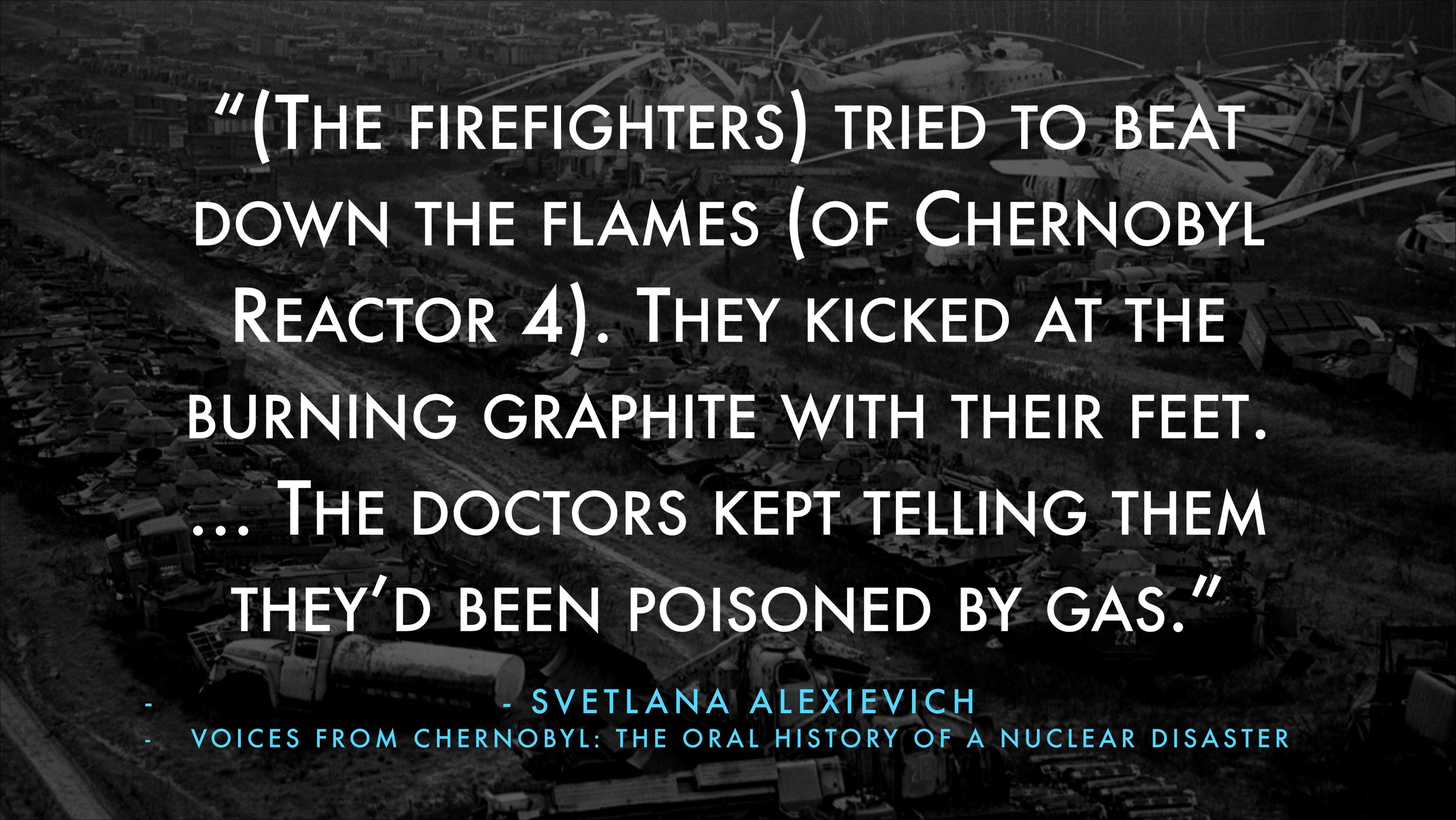
WHAT HAPPENED?

Case Study: Exchange Throttling

WE HIT AN IMPLICIT EXCHANGE LIMIT.
(ARGUABLY, A BUG.)

LESSONS LEARNED

IT IS POSSIBLE TO HAVE TOO
LITTLE INFORMATION.

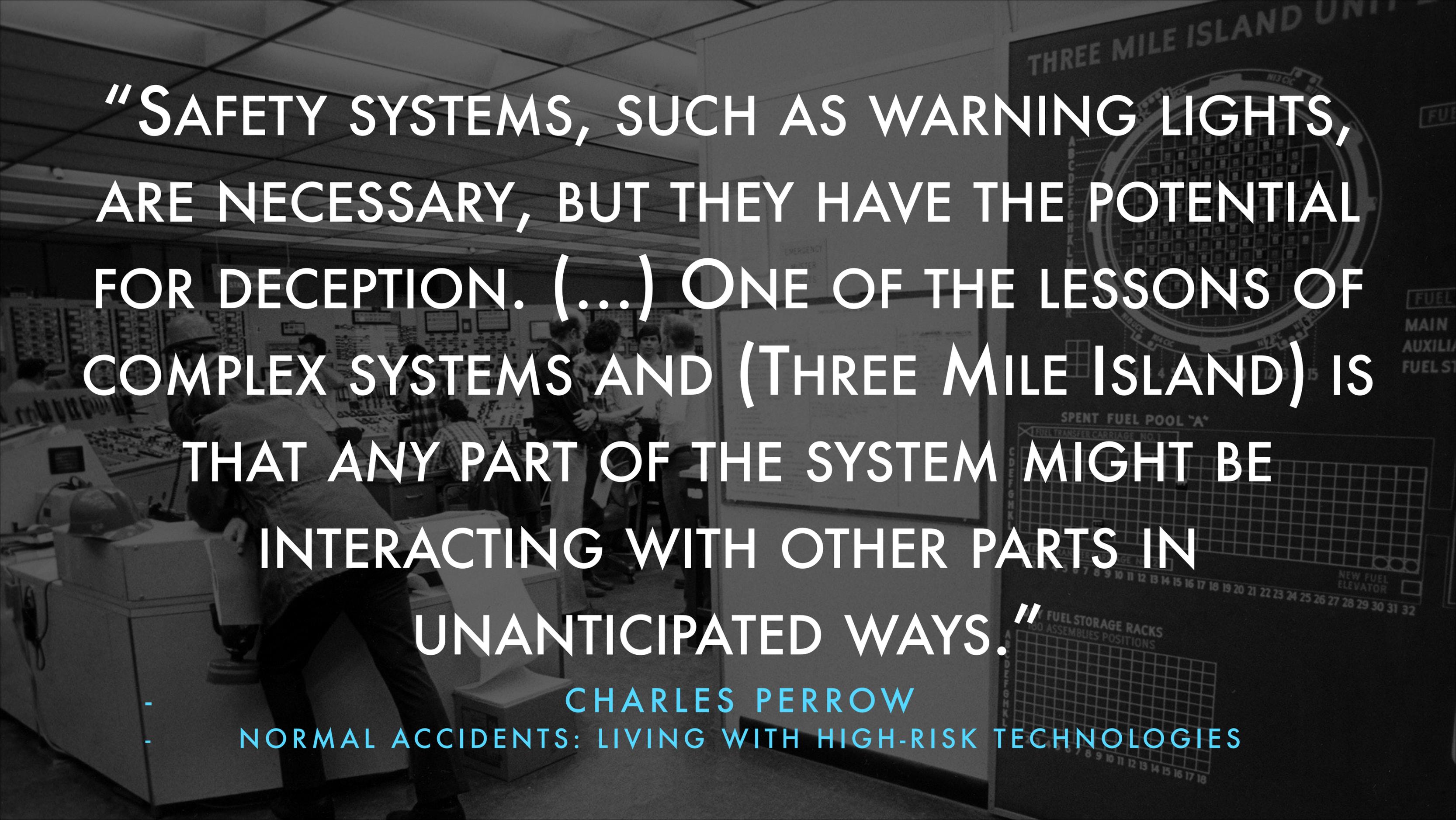


“(THE FIREFIGHTERS) TRIED TO BEAT
DOWN THE FLAMES (OF CHERNOBYL
REACTOR 4). THEY KICKED AT THE
BURNING GRAPHITE WITH THEIR FEET.
... THE DOCTORS KEPT TELLING THEM
THEY’D BEEN POISONED BY GAS.”

- SVETLANA ALEXIEVICH

- VOICES FROM CHERNOBYL: THE ORAL HISTORY OF A NUCLEAR DISASTER

IT IS POSSIBLE TO COLLECT TOO
MUCH INFORMATION, OR PRESENT
IT BADLY.

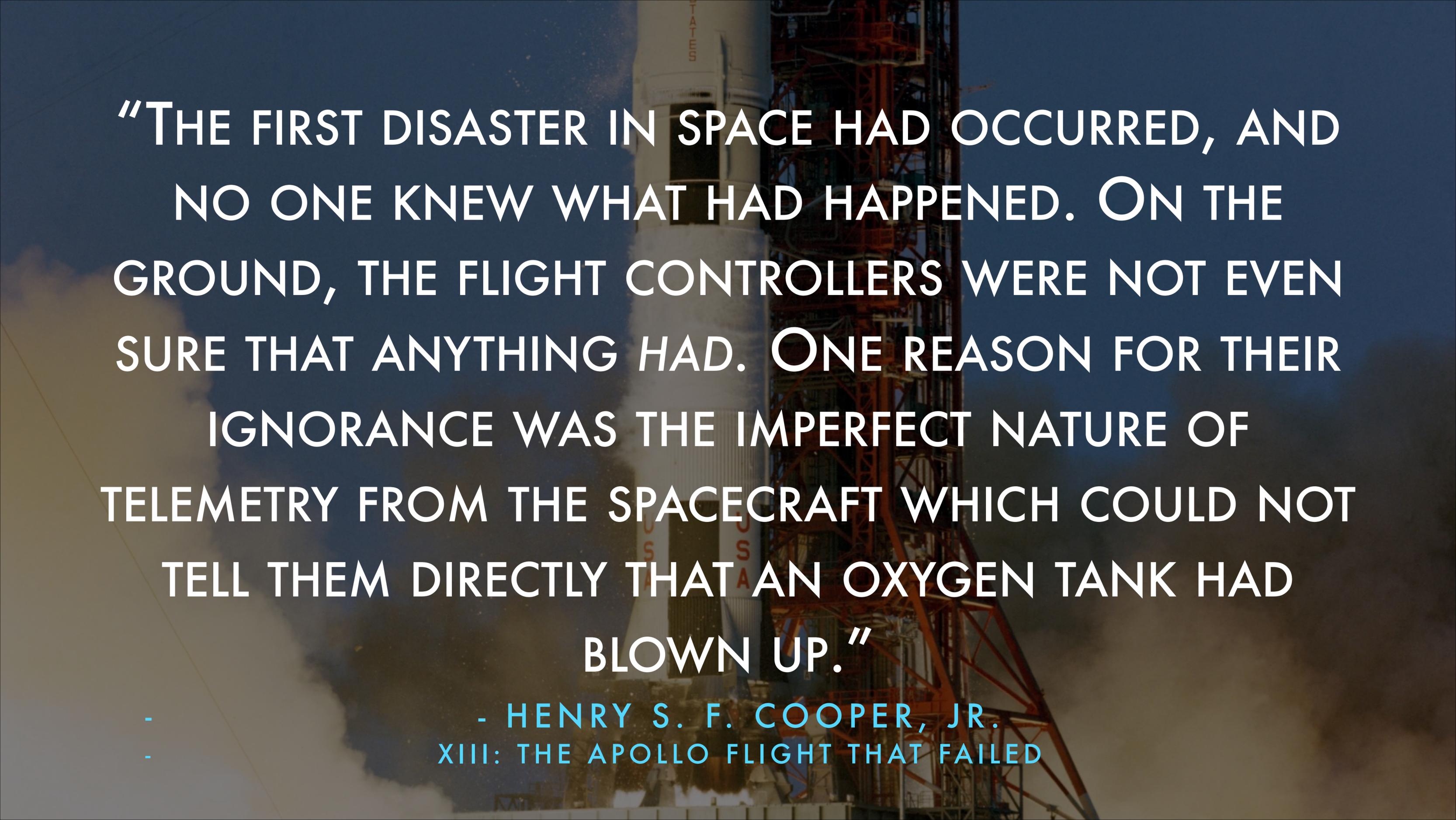


"SAFETY SYSTEMS, SUCH AS WARNING LIGHTS, ARE NECESSARY, BUT THEY HAVE THE POTENTIAL FOR DECEPTION. (...) ONE OF THE LESSONS OF COMPLEX SYSTEMS AND (THREE MILE ISLAND) IS THAT ANY PART OF THE SYSTEM MIGHT BE INTERACTING WITH OTHER PARTS IN UNANTICIPATED WAYS."

CHARLES PERROW

NORMAL ACCIDENTS: LIVING WITH HIGH-RISK TECHNOLOGIES

INDIRECT KNOWLEDGE MAY NOT TELL THE
WHOLE STORY, OR MAY MAKE YOU DOUBT
WHAT'S PLAINLY BEFORE YOUR EYES.



“THE FIRST DISASTER IN SPACE HAD OCCURRED, AND NO ONE KNEW WHAT HAD HAPPENED. ON THE GROUND, THE FLIGHT CONTROLLERS WERE NOT EVEN SURE THAT ANYTHING HAD. ONE REASON FOR THEIR IGNORANCE WAS THE IMPERFECT NATURE OF TELEMETRY FROM THE SPACECRAFT WHICH COULD NOT TELL THEM DIRECTLY THAT AN OXYGEN TANK HAD BLOWN UP.”

- HENRY S. F. COOPER, JR.

XIII: THE APOLLO FLIGHT THAT FAILED

Things that don't quite
work like you'd hope.



Problems

- Instrumenting code increases code size.
- While very low, runtime impact is not zero.
- Instrumentation of dependencies is up to the library author.

Solutions

- Dtrace / systemtap hookups
- Culture of instrumentation by default.
- Tracing BIFs

WOMBAT?

Instrumentation by default strategies

- Application configuration callback module
- Behaviors callbacks
- Value-at-time functions

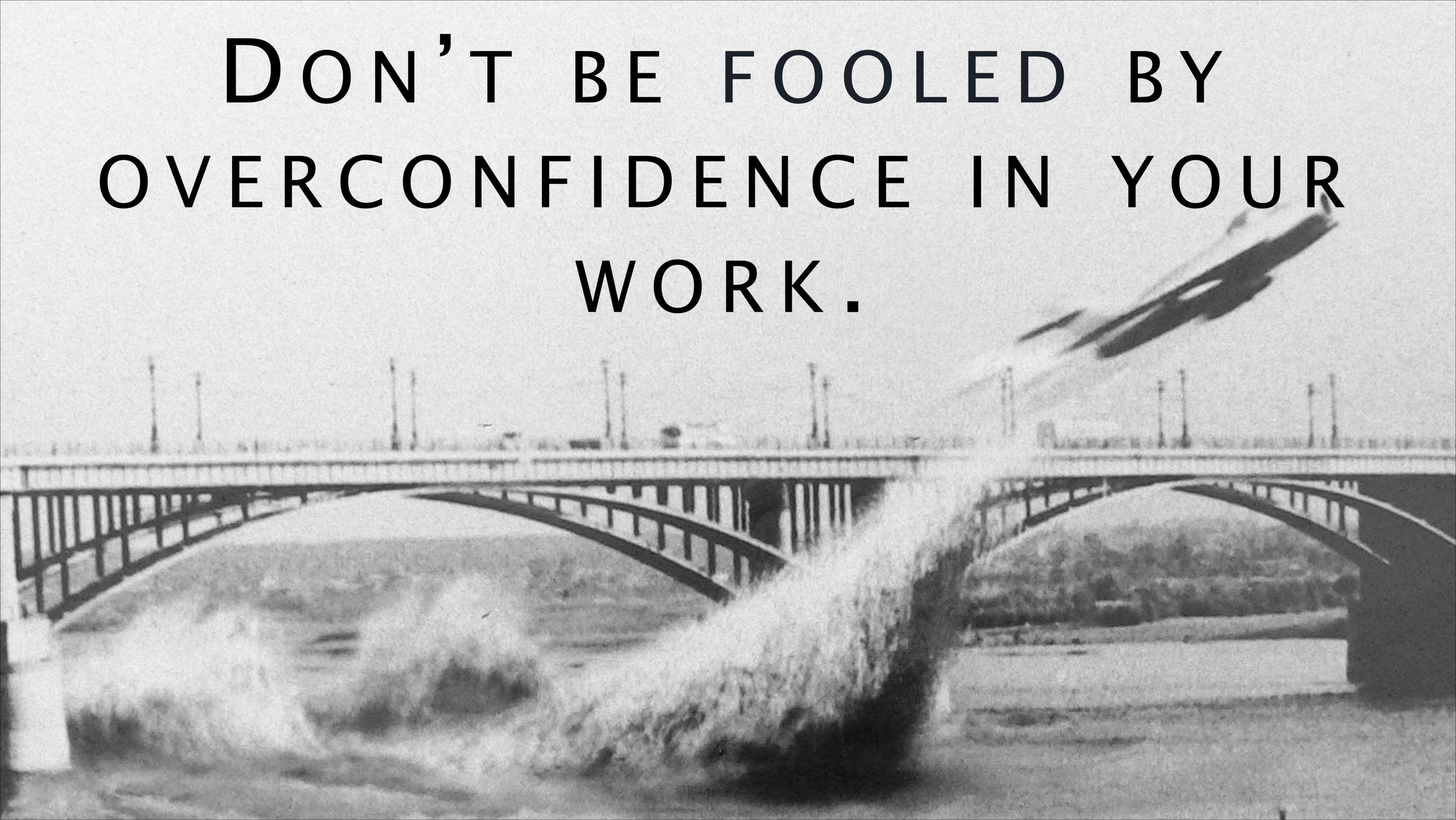


WHAT
ABOUT
CULTURE
E

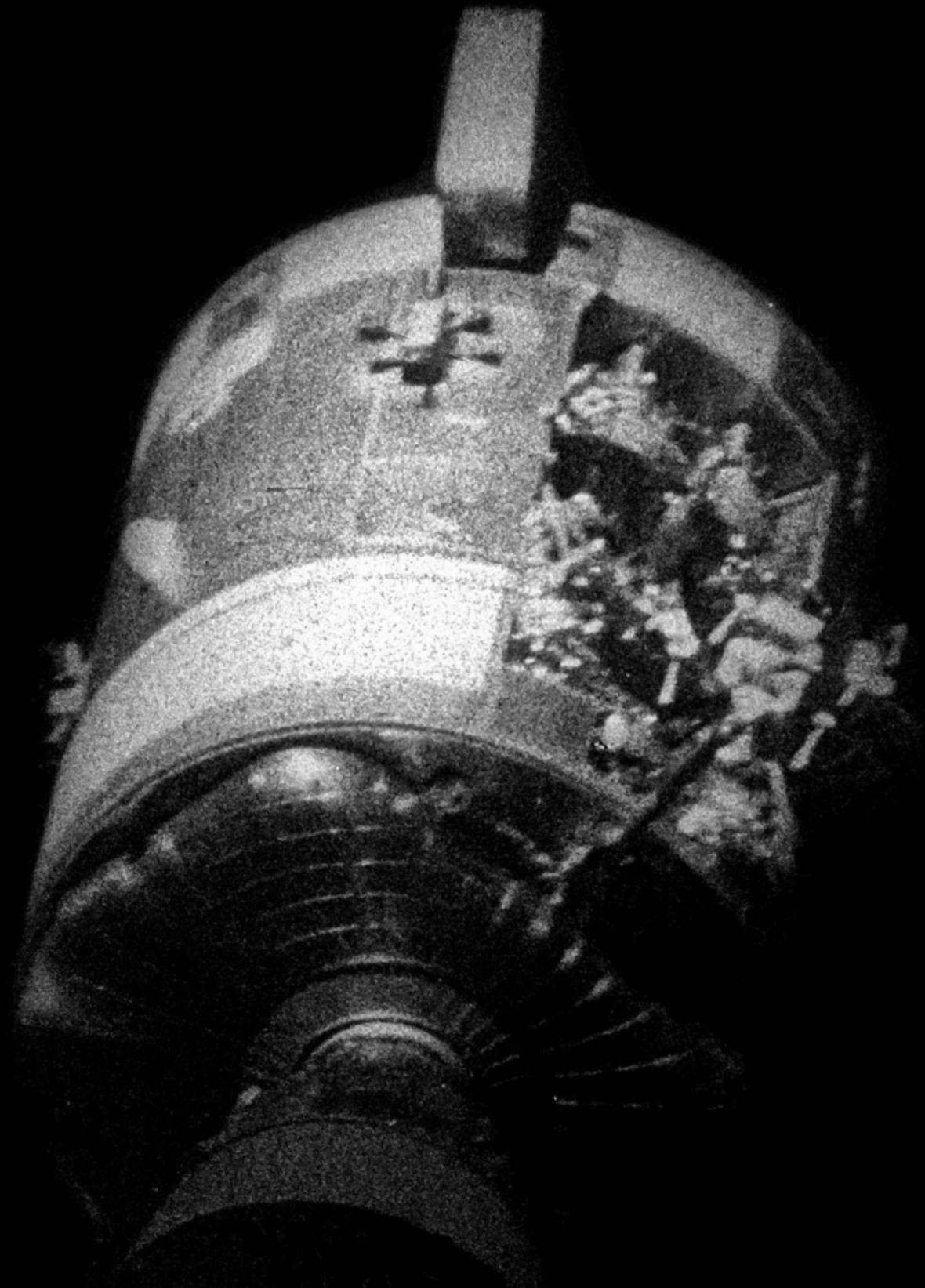


LOOK FOR
SOLUTIONS,
NOT
SCAPEGOATS.

DON'T BE FOOLED BY
OVERCONFIDENCE IN YOUR
WORK.



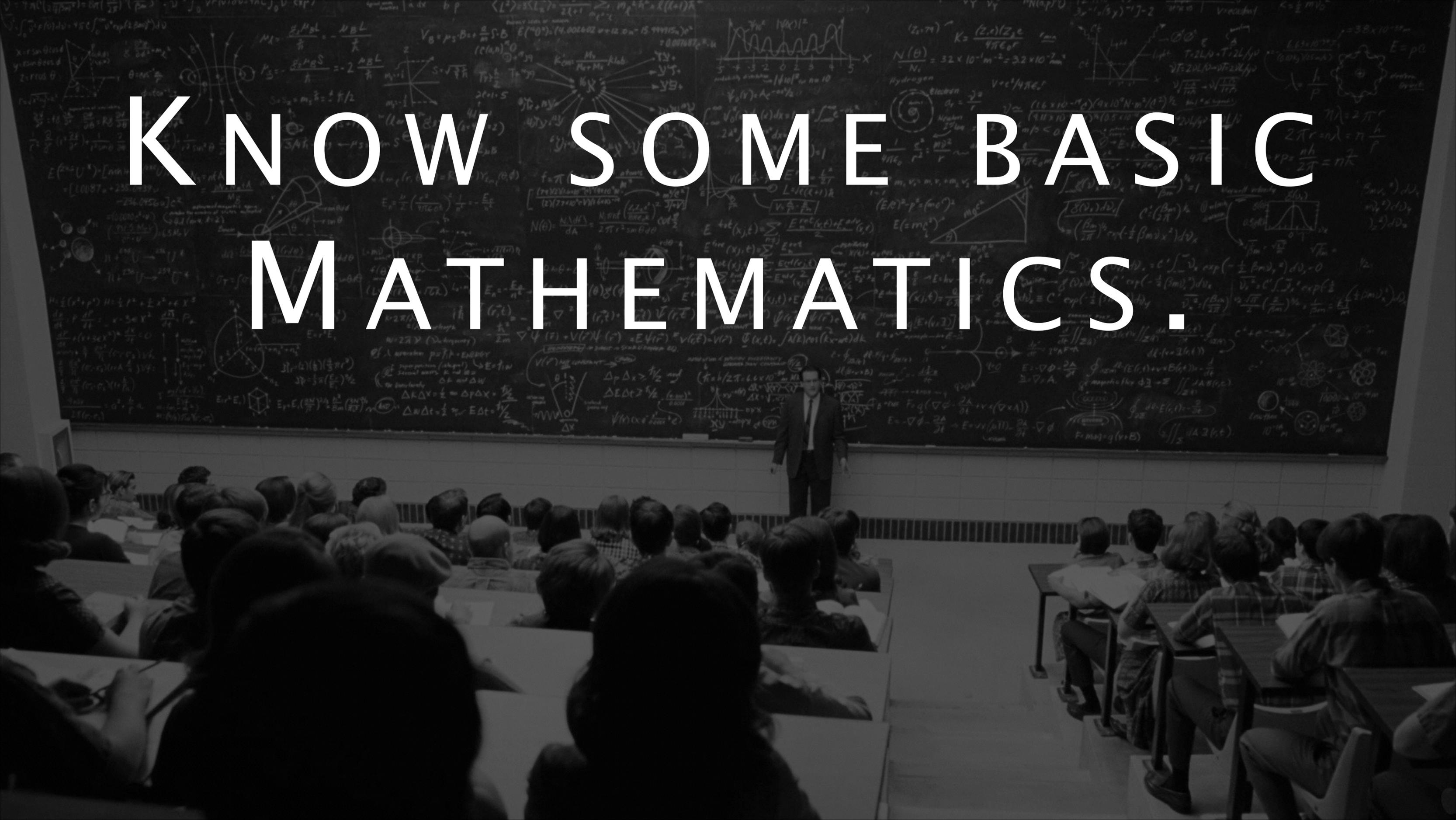
DOCUMENT
AND
DISCUSS
YOUR
FAILURES.





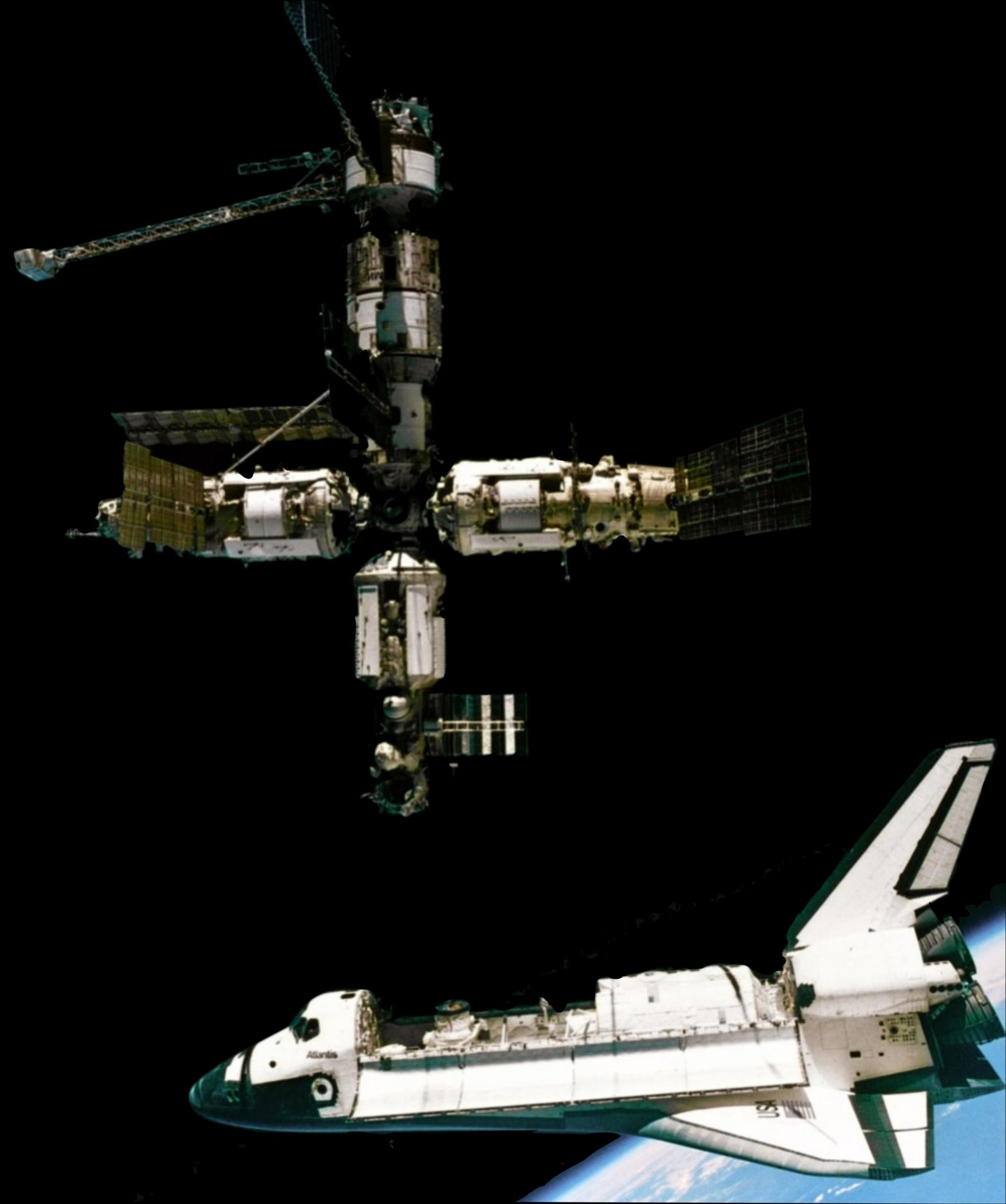
EVERYONE
MUST WORK
TOWARD THE
SAME END.

KNOW SOME BASIC MATHEMATICS.



PREFER LOOSE
COUPLING WHEN
POSSIBLE.

BE EXPLICIT
ABOUT TIGHT
COUPLING.





MEASUREMENT MUCH,
PRESENT ONLY THAT
WHICH YOU'RE
CERTAIN TO NEED.

MEASURE,
DON'T GUESS.



BE
ACCURATE

▪





QUESTIONS?

<3

@bltroutwine

BRIAN@TROUTWINE.US

XIII: The Apollo Flight that Failed
Cooper, Jr.

Henry S. F.

Digital Apollo: Human and Machine in Spaceflight
Mindell

David A.

Normal Accidents: Living with High-Risk Technologies
Charles Perrow

Voices from Chernobyl: The Oral History of a Nuclear Disaster

Svetlana Alexievich

Real-Time Systems: Design Principles for Distributed Embedded Applications

Hermann Kopetz

Command and Control: Nuclear Weapons, the Damascus Accident and the Illusion of Safety