

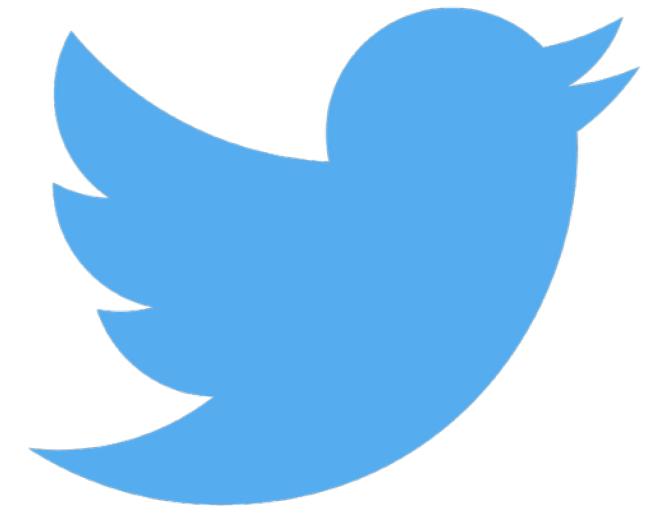


TOWARD BETTER DESIGN AND CODE

A case study with Pelikan Cache

ABOUT ME

- work at Twitter
- everything related to cache
- every job on the team



github.com/thinkingfish

[@thinkingfish](https://twitter.com/thinkingfish)

WHY BETTER DESIGN AND CODE?

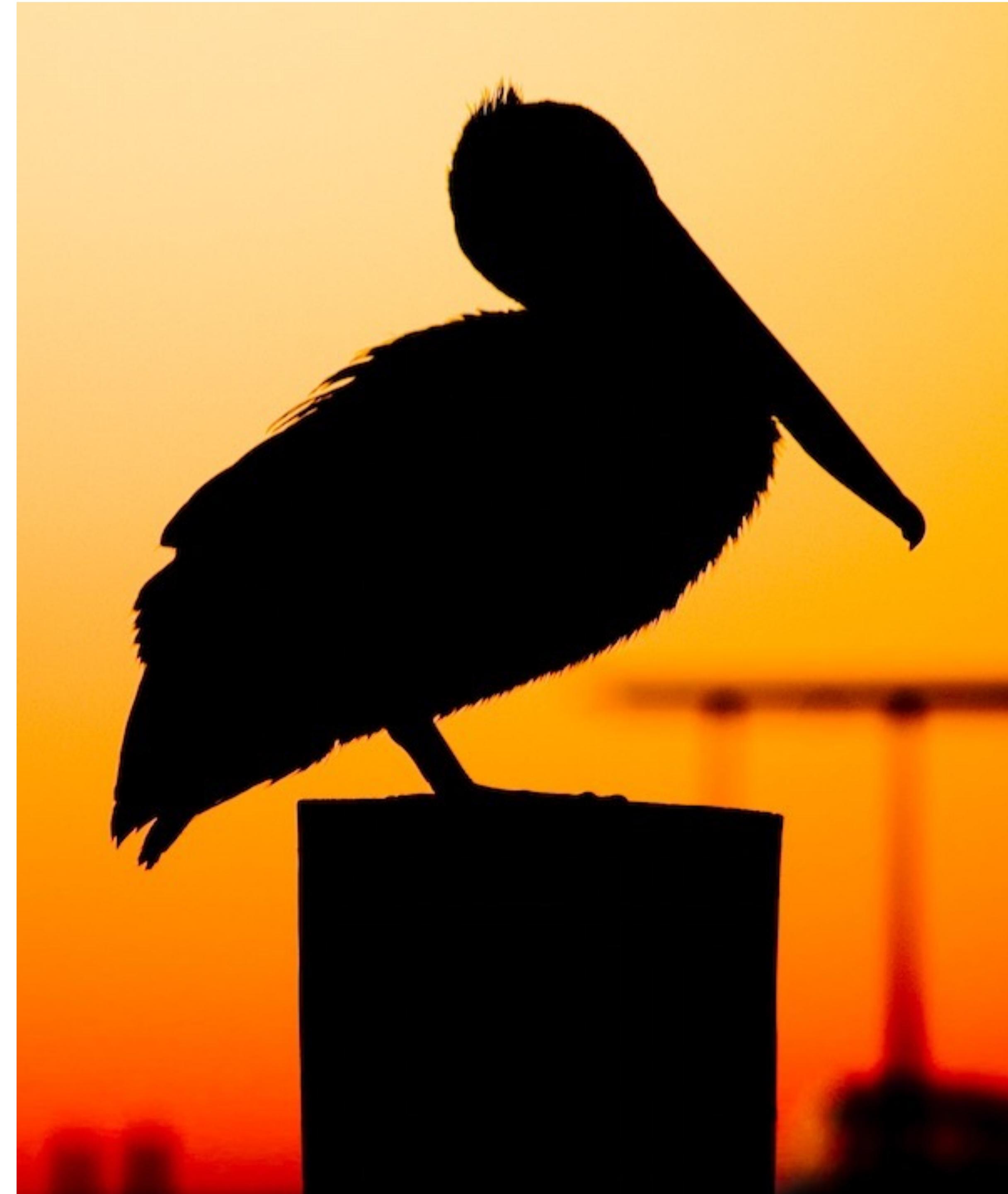
- Quality matters *...in the long term.*
- Joy matters
- This is Craft Conference, duh!

THE POWER OF STORIES



image by [Silvia Cordedda](#)

THE CASE

A high-contrast silhouette of a pelican in flight, its wings spread wide, set against a vibrant orange and yellow sunset. The bird is positioned on the left side of the frame, with its head turned slightly to the right. The background is a soft-focus gradient of warm colors.

THE PELIKAN PROJECT (PELIKAN.IO)

- Unify cache
- Years since inception
- “nothing new”
- set the stage for bigger ideas

THE PROCESS

DO WE DESERVE NICER THINGS?

The quest for “better”.

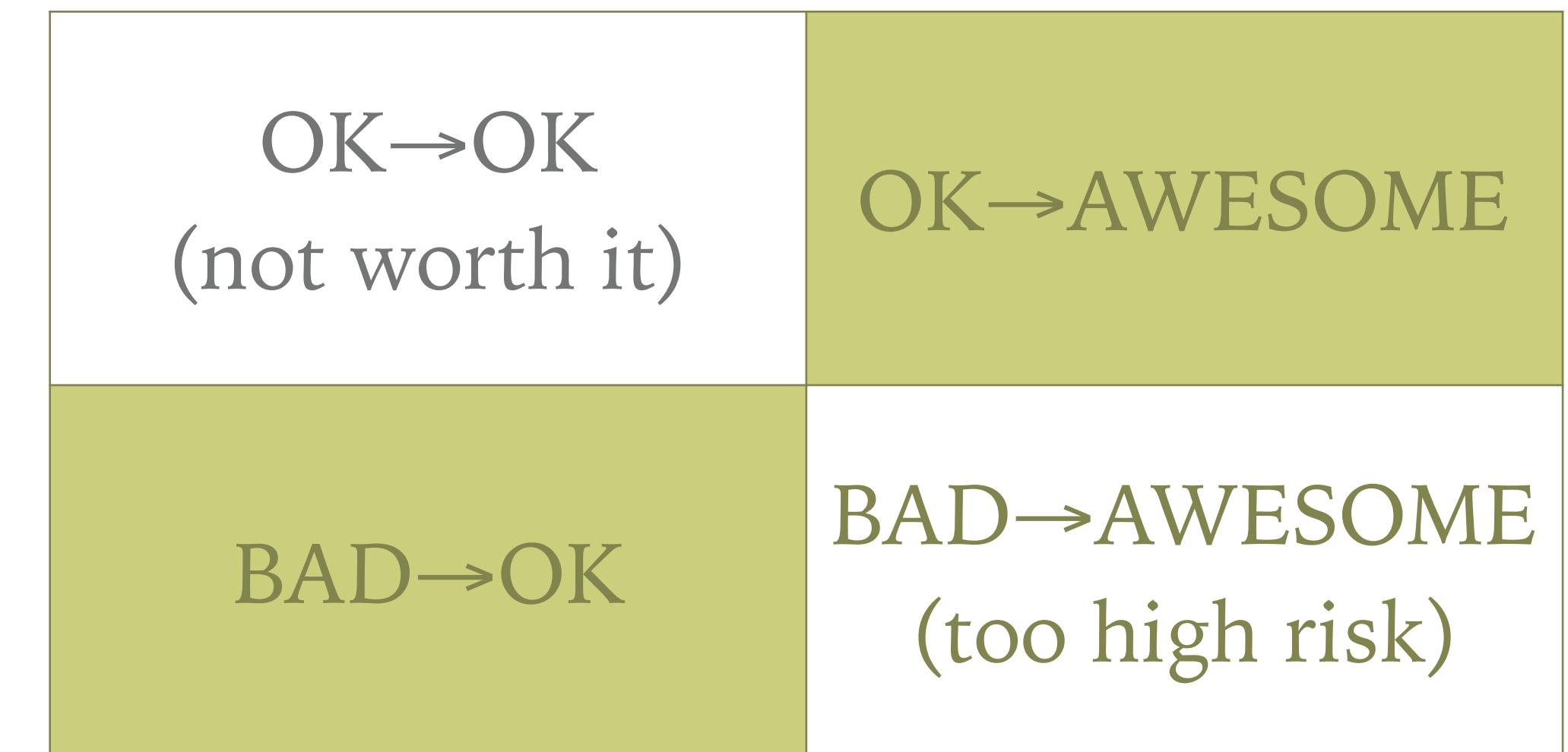
“

Better is the enemy of good.

-Voltaire

REALITY

- We cannot always afford “better”.
- Good needs to happen first.
- Timing matters.



*Ryan King's 2x2 risk matrix
“evolution, not revolution”*

EVALUATION

- Is it worth it?
- What's important?
- What does it take?

INSIGHT PRINCIPLE STYLE

The execution toward “better”.

WANT INSIGHT? ASK QUESTIONS.

INTELLIGENCE

Not because you think you know everything without questioning, but rather because you question everything you think you know.



“

The \$64,000 Question

an American game show, 1950s

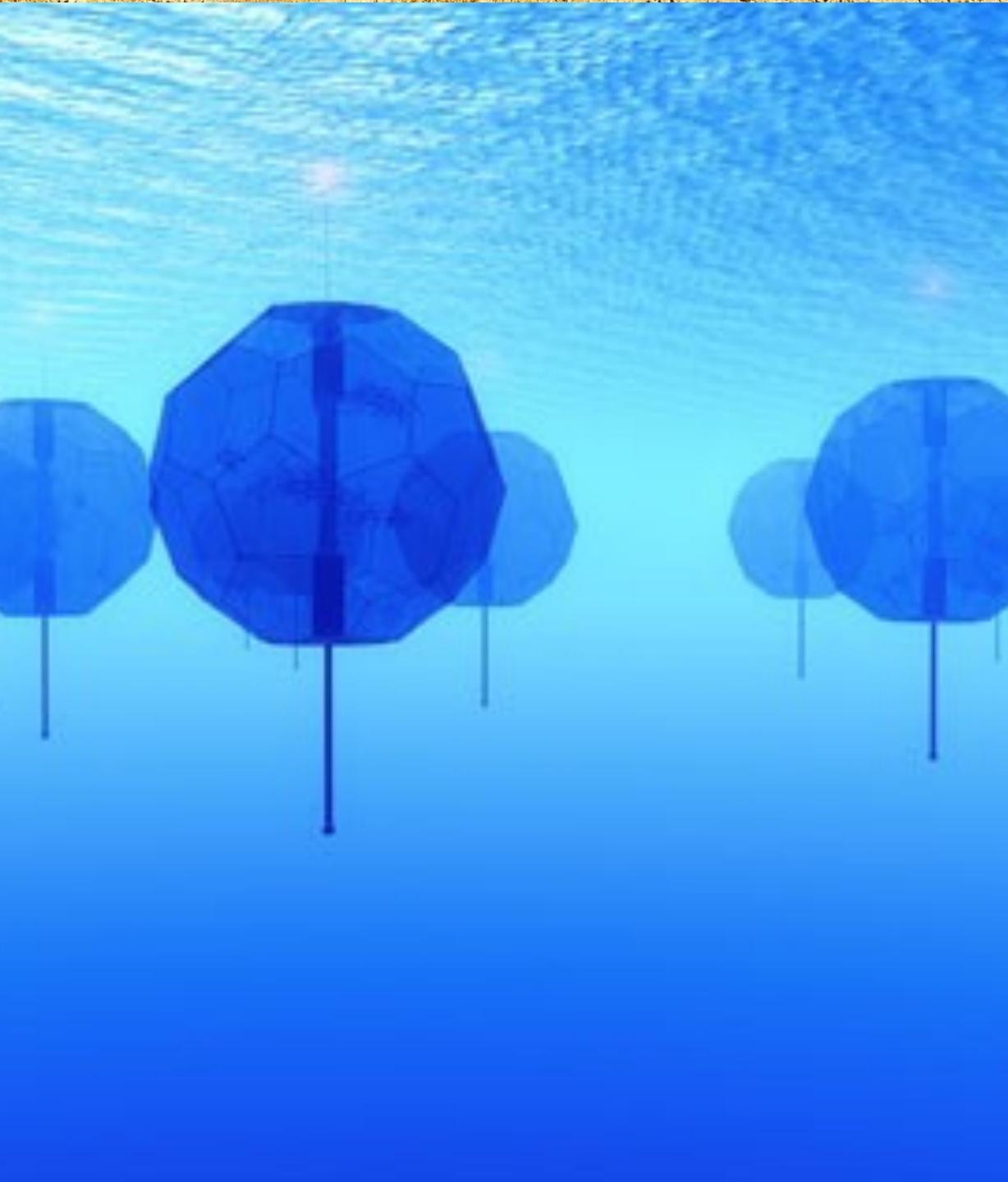
QUESTIONS WE ASKED

- What is caching?
- How do people use caches?
- What performance is acceptable?
- What causes developers pain?
- What makes debugging hard?
- What prevents us from adding new features?

“

Form is liberating.

-Artists' Aphorism



CONTEXT, CONTEXT, CONTEXT

QUESTIONS EASIER TO ANSWER

- What is caching *in today's datacenter*?
- How do *Twitter's microservices* use caches?
- What performance is acceptable *for real-time web services*?
- What causes *app/web service owners* pain?
- What makes debugging hard in *container-based environment*?
- What prevents us from adding *{transports, protocols, datastore}*?

Production-ready

GOAL: A ~~BETTER~~ CACHE BACKEND

- Predictable runtime
- Operations friendly
- Maintainable

Production-ready

GOAL: A ~~BETTER~~ CACHE BACKEND

- Predictable runtime
 - throughput / latencies *under load*
 - CPU / memory footprint
- Operations friendly
 - capable configuration
 - good observability
- Maintainable
 - easy to add functionality
 - easy to change implementation

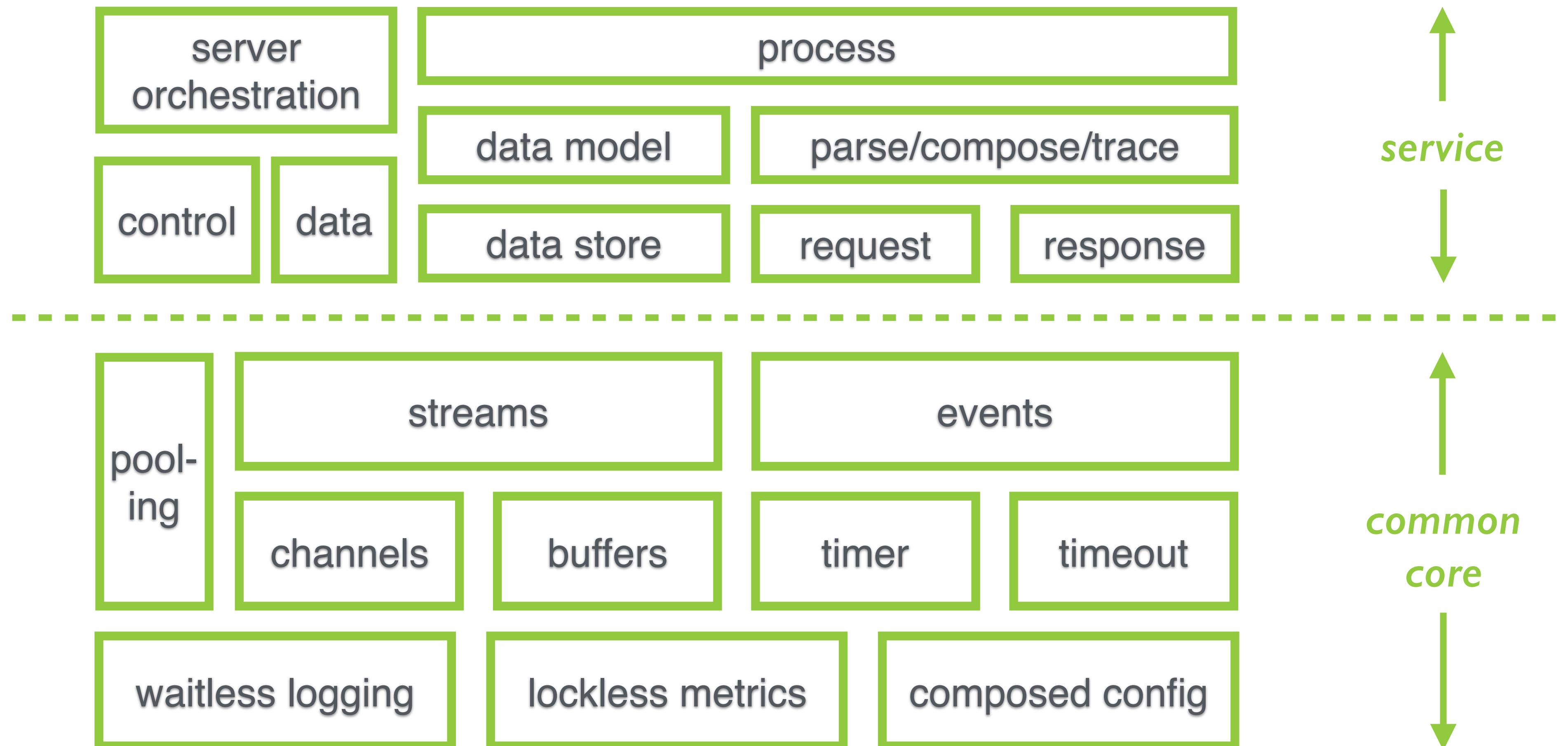
INSIGHT PRINCIPLE STYLE

The execution toward “better”.

WELL-GROOMED INTERFACE



THE PELIKAN MODULES





JUST-ENOUGH ABSTRACTION

FUTURE-PROOFING: MINIMUM OBSTRUCTION





VERTICAL INTEGRATION

“

God is in the detail.

-old proverb

CORNER CASES



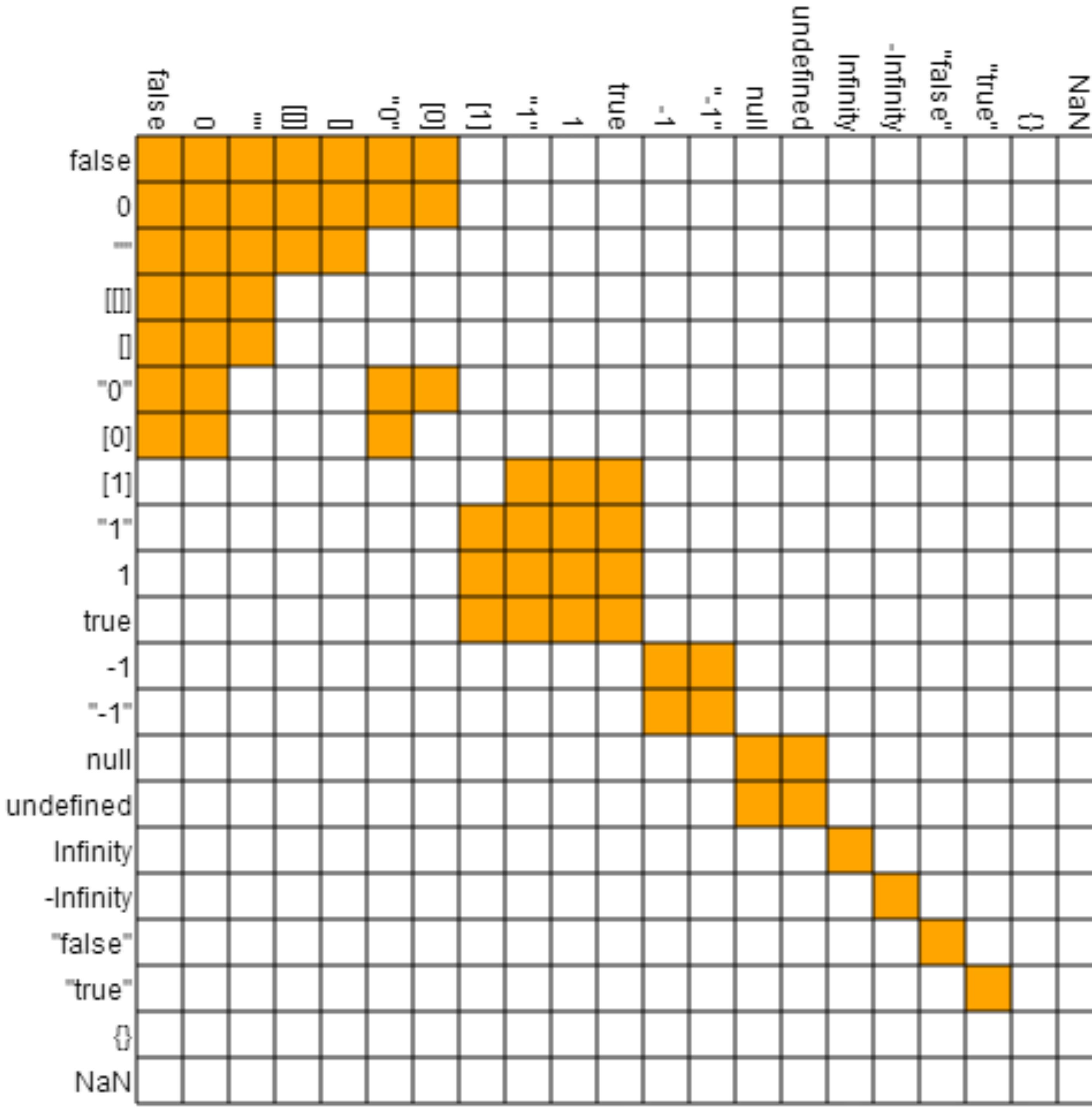
“

There are only two hard things in Computer
Science: cache invalidation and naming things.

-Phil Karlton

INSIGHT PRINCIPLE STYLE

The execution toward “better”.



**BE OBVIOUS,
BE CONSISTENT,
BE SIMPLE**

image by [Craig Gidney](#)

```
time_setup();

timing_wheel_setup(&glob_stats.timing_wheel_metrics)
;

procinfo_setup(&glob_stats.procinfo_metrics);
request_setup(&glob_stats.request_metrics);
response_setup(&glob_stats.response_metrics);
parse_setup(&glob_stats.parse_req_metrics, NULL);
compose_setup(NULL,
&glob_stats.compose_rsp_metrics);
klog_setup(setting.klog_file.val.vstr,
           setting.klog_backup.val.vstr,
           (uint32_t)setting.klog_nbuf.val.vuint,
           (uint32_t)setting.klog_intvl.val.vuint,
           (uint32_t)setting.klog_sample.val.vuint,
           (size_t)setting.klog_max.val.vuint,
           &glob_stats.klog_metrics);
process_setup(setting.allow_flush.val.vbool,
             &glob_stats.process_metrics);
admin_process_setup(&glob_stats.admin_process_metrics);
buf_setup((uint32_t)setting.buf_init_size.val.vuint,
&glob_stats.buf_metrics);
dbuf_setup((uint32_t)settingdbuf_max_power.val.vuin
t);
```

```
/* setup library modules */

buf_setup(&setting.buf, &stats.buf);
dbuf_setup(&settingdbuf, &statsdbuf);
event_setup(&stats.event);
sockio_setup(&setting.sockio);
tcp_setup(&setting.tcp, &stats.tcp);
timing_wheel_setup(&stats.timing_wheel);

/* setup pelikan modules */

time_setup();
procinfo_setup(&stats.procinfo);
request_setup(&setting.request, &stats.request);
response_setup(&setting.response, &stats.response);
parse_setup(&stats.parse_req, NULL);
compose_setup(NULL, &stats.compose_rsp);
klog_setup(&setting.klog, &stats.klog);
slab_setup(&setting.slab, &stats.slab);
process_setup(&setting.process, &stats.process);
admin_process_setup(&stats.admin_process);
core_setup(&setting.admin, &setting.server,
&setting.worker,
&stats.server, &stats.worker);
```

```
event_setup(&glob_stats.event_metrics);
tcp_setup((int)setting.tcp_backlog.val.vuint, &glob_stats.tcp_metrics);
if (slab_setup((uint32_t)setting.slab_size.val.vuint,
               setting.slab_prealloc.val.vbool,
               (int)setting.slab_evict_opt.val.vuint,
               setting.slab_use_freeq.val.vbool,
               (size_t)setting.slab_min_chunk_size.val.vuint,
               (size_t)setting.slab_max_chunk_size.val.vuint,
               (size_t)setting.slab_maxbytes.val.vuint,
               setting.slab_profile.val.vstr,
               setting.slab_profile_factor.val.vstr,
               &glob_stats.slab_metrics)
        != CC_OK) {
    log_error("slab module setup failed");
    goto error;
}

if (item_setup(setting.item_use_cas.val.vbool,
               (uint32_t)setting.item_hash_power.val.vuint,
               &glob_stats.item_metrics)
        != CC_OK) {
    log_error("item setup failed");
    goto error;
}
```

```
buf_sock_pool_create((uint32_t)setting.buf_sock_poolsizes.val.vuint);
request_pool_create((uint32_t)setting.request_poolsizes.val.vuint);
response_pool_create((uint32_t)setting.response_poolsizes.val.vuint);
status = getaddr(&data_ai, setting.server_host.val.vstr,
                 setting.server_port.val.vstr);
if (status != CC_OK) {
    log_error("server address invalid");
    goto error;
}
status = getaddr(&admin_ai, setting.admin_host.val.vstr,
                 setting.admin_port.val.vstr);
if (status != CC_OK) {
    log_error("admin address invalid");
    goto error;
}
max_conns = setting.tcp_poolsizes.val.vuint == 0 ?
    setting.ring_array_cap.val.vuint : setting.tcp_poolsizes.val.vuint;
status = core_setup(data_ai, admin_ai, max_conns,
    (int)setting.admin_intvl.val.vuint, setting.admin_tw_tick.val.vuint,
    setting.admin_tw_cap.val.vuint, setting.admin_tw_ntick.val.vuint,
    &glob_stats.server_metrics, &glob_stats.worker_metrics);
freeaddrinfo(data_ai);
freeaddrinfo(admin_ai);
```

SIMPLE RULES FOR COMPLEXITY



```
#include "setting.h"

struct setting setting = {
    { TWEMCACHE_OPTION(OPTION_INIT) }, ,
    { ADMIN_OPTION(OPTION_INIT) }, ,
    { SERVER_OPTION(OPTION_INIT) }, ,
    { WORKER_OPTION(OPTION_INIT) }, ,
    { PROCESS_OPTION(OPTION_INIT) }, ,
    { KLOG_OPTION(OPTION_INIT) }, ,
    { REQUEST_OPTION(OPTION_INIT) }, ,
    { RESPONSE_OPTION(OPTION_INIT) }, ,
    { SLAB_OPTION(OPTION_INIT) }, ,
    { ARRAY_OPTION(OPTION_INIT) }, ,
    { BUF_OPTION(OPTION_INIT) }, ,
    { DBUF_OPTION(OPTION_INIT) }, ,
    { DEBUG_OPTION(OPTION_INIT) }, ,
    { SOCKIO_OPTION(OPTION_INIT) }, ,
    { TCP_OPTION(OPTION_INIT) }, ,
};

unsigned int nopt = OPTION_CARDINALITY(struct
setting);
```

```
#include "stats.h"

struct stats stats = {
    { PROCINFO_METRIC(METRIC_INIT) }, ,
    { PROCESS_METRIC(METRIC_INIT) }, ,
    { ADMIN_PROCESS_METRIC(METRIC_INIT) }, ,
    { PARSE_REQ_METRIC(METRIC_INIT) }, ,
    { COMPOSE_RSP_METRIC(METRIC_INIT) }, ,
    { KLOG_METRIC(METRIC_INIT) }, ,
    { REQUEST_METRIC(METRIC_INIT) }, ,
    { RESPONSE_METRIC(METRIC_INIT) }, ,
    { SLAB_METRIC(METRIC_INIT) }, ,
    { CORE_SERVER_METRIC(METRIC_INIT) }, ,
    { CORE_WORKER_METRIC(METRIC_INIT) }, ,
    { BUF_METRIC(METRIC_INIT) }, ,
    { DBUF_METRIC(METRIC_INIT) }, ,
    { EVENT_METRIC(METRIC_INIT) }, ,
    { LOG_METRIC(METRIC_INIT) }, ,
    { TCP_METRIC(METRIC_INIT) }, ,
    { TIMING_WHEEL_METRIC(METRIC_INIT) }, ,
};

unsigned int nmetric = METRIC_CARDINALITY(stats);
```

KEEP IT ALIVE

Stay “better”.



THE POWER OF PROTOTYPE

THE MYTH CALLED CODE REUSE





REFACTOR, REFACTOR, REFACTOR

Additions and Deletions per week



EXISTENTIAL CRISIS, NIH SYNDROME





CRAFTSMANSHIP IS FOREVER



image by [Silvia Cordedda](#)