Twitter Thread by Ben Sigelman

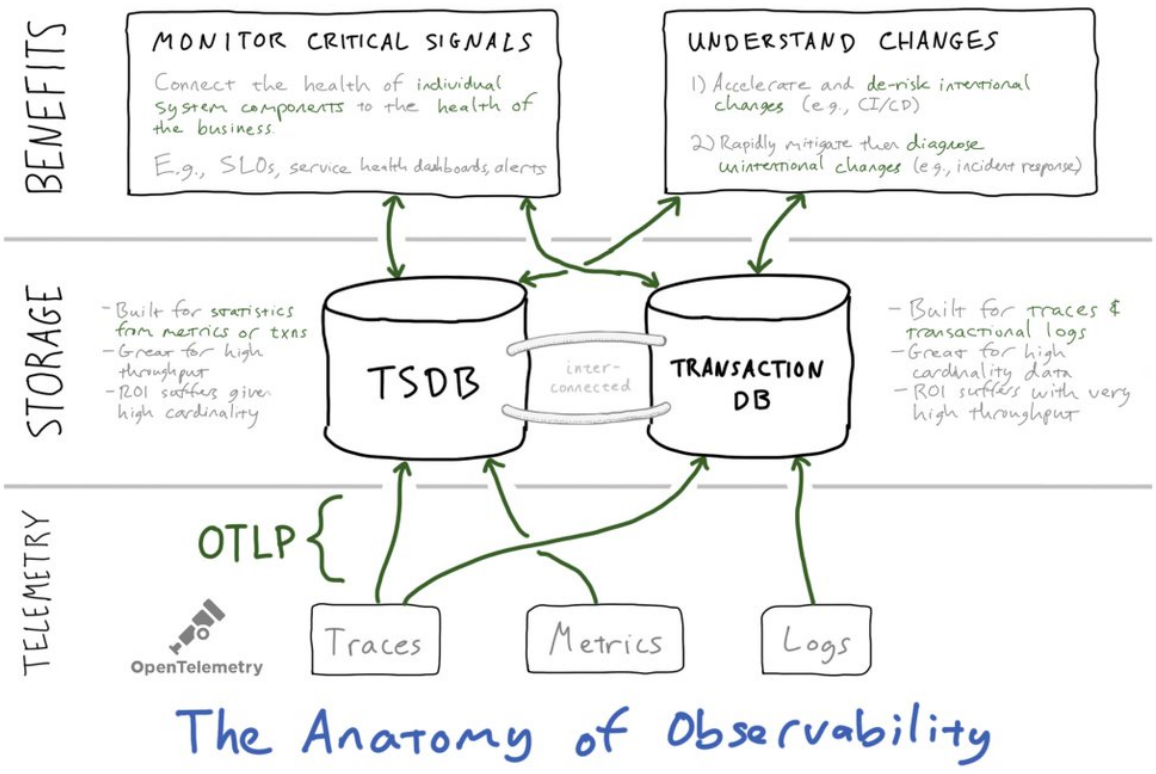


Ben Sigelman
@el_bhs

0/ I'm tired of hearing about observability replacing monitoring. It's not going to, and that's because it shouldn't.

Observability will not _replace_ monitoring, it will _augment_ monitoring.

Here's a thread about observability, and how monitoring can evolve to fit in: ■



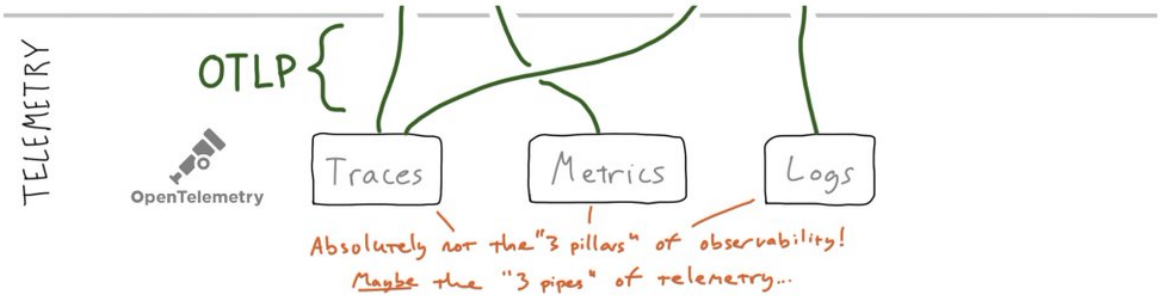
1/ Let's start with the diagram (above) illustrating the anatomy of observability. There are three layers:

I. (Open)Telemetry: acquire high-quality data with minimal effort

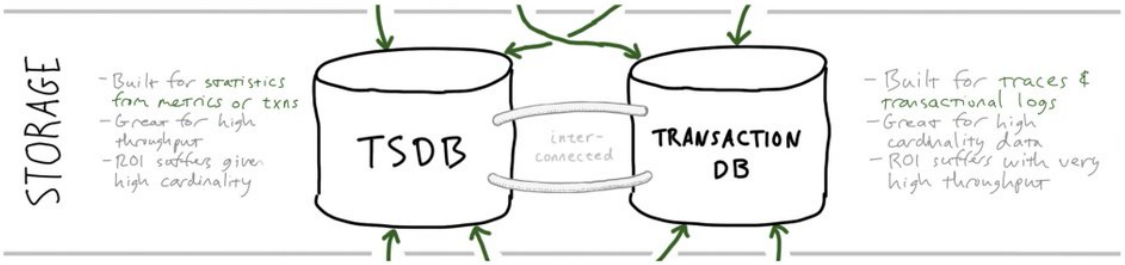
II. Storage: “Stats over time” and “Transactions over time”
III. Benefits: *solve actual problems*

2/ The direction for “Telemetry” is simple: @opentelemetry.

(This is the (only) place where the so-called "three pillars” come in, by the way. If you think you’ve solved the observability problem by collecting traces, metrics, and logs, you’re about to be disappointed. :-/)



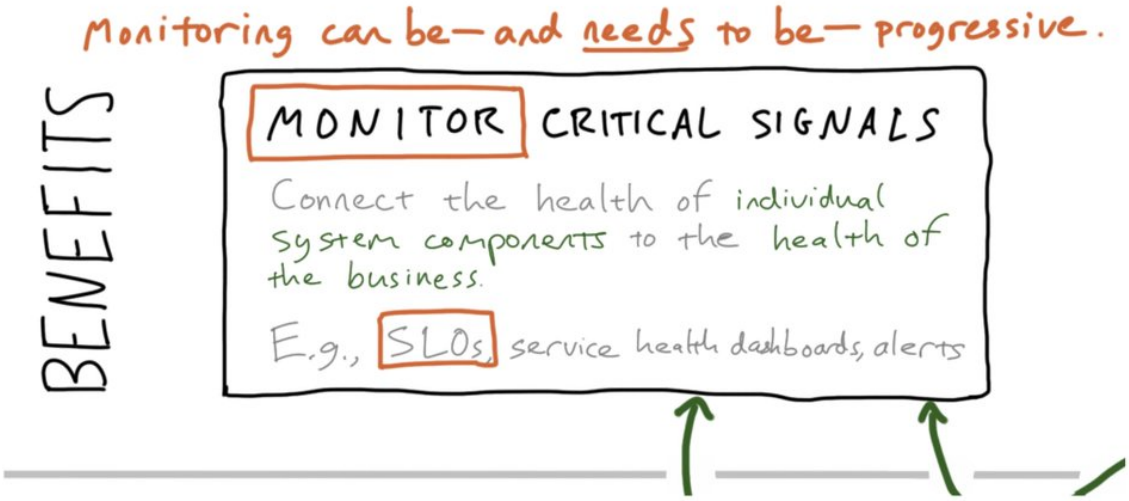
3/ The answer for “Storage” depends on your workload, but we’ve learned that it’s glib to expect a data platform to support observability with *just* a TSDB or *just* a transaction/trace/logging DB. And also that “cost profiling and control” is a core platform feature.



4/ But what about “Benefits”? There’s all of that business about Control Theory (too academic) and “unknown unknowns” (too abstract). And “three pillars” which is complete BS, per the above (it’s just “the three pillars of telemetry,” at best).

5/ Really, Observability *Benefits* divide neatly into two categories: understanding *health* (i.e., monitoring) and understanding *change* (i.e., finding and exposing signals and statistical insights hidden within the firehose of telemetry).

6/ Somewhere along the way, “monitoring” was thrown under a bus, which is unfortunate. If we define monitoring as *an effort to connect the health of a system component to the health of the business* – it’s actually quite vital. And ripe for innovation! E.g., SLOs.



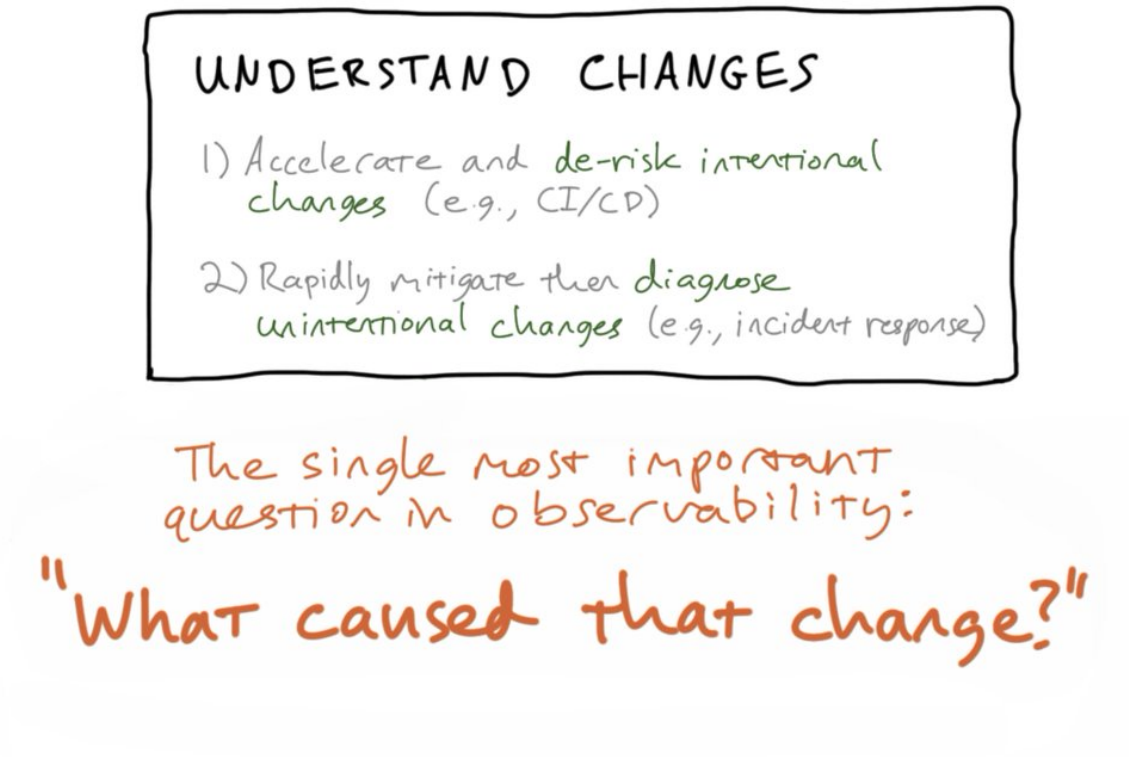
7/ “Monitoring” got a bad name because operators were *trying to monitor every possible failure mode of a distributed system.* That doesn’t work because there are too many of them.

(And that’s why you have too many dashboards at your company.)

8/ Monitoring doesn’t have to be that way. It can actually be quite clarifying, and there’s still ample room for innovation. I’d argue that SLOs, done properly, are what monitoring can and should be (or become).

9/ So what if we do things differently? What if we do things *right*? We treat Monitoring as a first-class citizen, albeit only one aspect of observability, and we closely track the signals that best express and predict the health of each component in our systems.

10/ … And then we need a new kind of observability value that's purpose-built to manage *changes* in those signals. More on that part in a future post. :) But the idea is to facilitate intentional change (e.g., CI/CD) while mitigating unintentional change (Incident Response).



11/ Zooming out: Monitoring will never be *replaced* by Observability: it's not just "part of Observability's anatomy," it's a vital organ! Our challenge is to *evolve* Monitoring, and to use it as a scaffold for the patterns and insights in our telemetry that explain change.