



# Observability for **Embedded Devices**

# We make software the **most reliable** part of the IoT

We believe that observability, purpose built for embedded devices, gives developers the visibility they need to build super sophisticated, highly reliable products in even the most constrained environments.



# We were **busy** in 2023

## More support for more platforms

- More out-of-the-box Android metrics
- More control for Android OTA Updates
- Improved support for Qualcomm based devices
- Improved out-of-the-box support for more Linux distro's
- Session metrics and compact logging on MCU
- Support for Cortex-R based chips

## More insights, more easily

- Completely customizable dashboards
- Fleet-wide reboot reason tracking
- Real-time data visibility in metric charts
- Multiple new chart types for metric data analysis
- Improved Issue management and triaging
- Increased configurability and control for alerting

And we have some more **exciting**  
**news** to share

But first, **some context**

# Building devices is **hard**



The ecosystem for building embedded devices is fragmented.



Off-the-shelf doesn't really exist and every device is unique.



You have to write software to make everything work together and deliver a good quality experience.



All within the constraints of intermittent connectivity, limited compute and limited power.

# Development takes time

## Things to know:

**16 Months**

Average development lifecycle

**30%**

of active projects are behind schedule

## Reasons for delay:

**23%**

Technical obstacles

**30%**

complexity of the application or technology

# The pressure **is high**

**56**

Average Engineering and Project Management **staff per Project**

**\$4MM**

Average Cost per **Development Project**

**29%**

Average percent of total spend on **Embedded Software**

**21%**

Average percent of total spend on **Electronic Engineering**

**18%**

Average percent of total spend on **Mechanical Engineering**



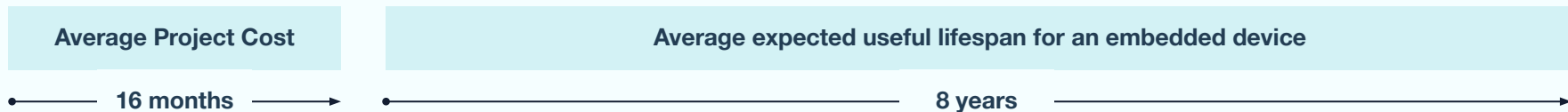
# And it **doesn't stop** at launch

**\$4MM**

Average Project Cost

**\$??MM**

Average lifetime maintenance cost



# Quality problems can have a **big impact**

<b>Measurable costs</b>	RMA Cost	Cost for an Engineer site visit	Customer Churn Cost
<b>Unseen costs</b>	Brand damage	Loss of future revenue	Staff stress and dissatisfaction

# We don't know **how to measure it**

## Cloud Deployments:

- Uptime ✓
- Error Rate ✓
- Latency ✓
- MTBF ✓

## Embedded Devices:

- 🤔
- 🤔
- 🤔
- 🤔

# Measures we have are **imperfect**

## Sales

- Sales growth
- Profitability

## Support

- Number of tickets
- Spend on support

## Returns

- Rate of returns
- Cost of returns

## Feedback

- NPS surveys
- Customer interviews

---

## Reviews

◆◆◆◆◆

“I just installed an update and now it’s constantly crashing...this sucks. DO NOT WASTE YOUR MONEY ON THIS”

◆◆◆◆◆

“They said the battery is supposed to last 2 days but I am having to charge mine every night.”

◆◆◆◆◆

“The bluetooth constantly disconnects making these unusable.”

# Even if you can get data... **what then?**

Does the data actually  
tell you anything?

Can you use it to  
understand what's  
actually happening  
quickly?

Can you respond?

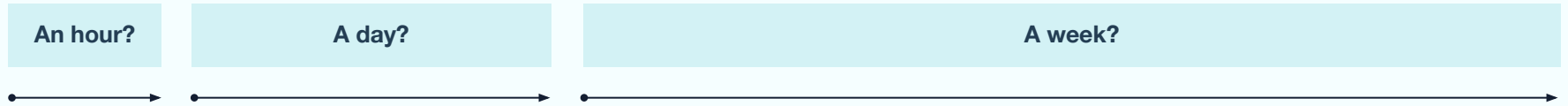
# What if I asked **you**...

How often does  
your customer  
experience a  
crash in the field?

How long is your  
battery life in the field,  
across all deployed  
devices?

How often do  
your devices fail  
to connect?

# How **long** would it take?



And how confident would you be?

# And could you **explain why?**

Why is battery life getting worse?

When did it start getting worse?

Did we change anything?



What if it took **60 seconds**...

And confidence in the data  
was so high you could send  
it straight to your boss.

**Or your CEO.**

And you could go from fleet  
wide data to an individual  
device **in moments.**

# DeviceVitals



# How stable is my firmware in the field?



Have I made  
improvements  
in my **latest**  
**release?**



And are things getting **better overtime?**



Across  
**every device**  
in my fleet...



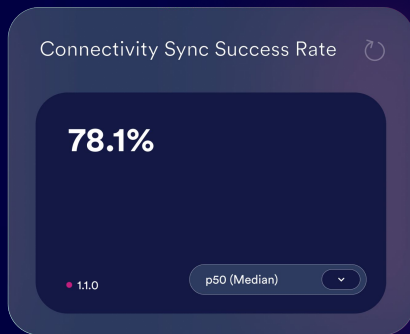


# For the most **critical** measures

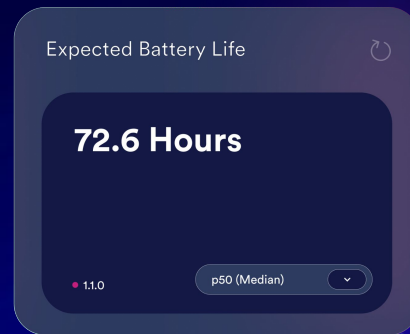
## Stability



## Connectivity



## Battery



# Device Vitals

## Stability

Crash-free hours

## Connectivity

Sync Success or Uptime

## Battery

Expected Battery Life

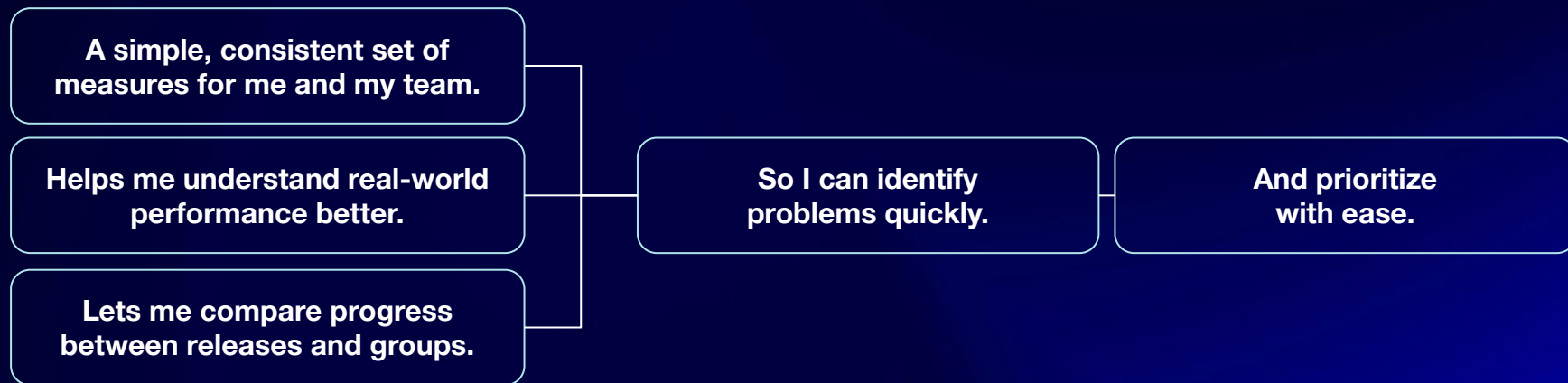
**Built-in on all  
SDKs**

**Available for  
every device**

**Deploy at any  
lifecycle state**

**All calculated  
automatically**

# Device Vitals



# How does it work?

All metrics are built-in on all SDKs across MCU, Linux and Android

Once data is collected, each Vital is computed automatically by Memfault

Purpose built charts and cards available out-of-the-box

## SDK

Collecting the metrics that inform your Device Vital signs.

Sync\_failure

Sync\_successful

Connectivity\_expected\_time\_ms

Connectivity\_connected\_time\_ms

Operational\_hours

Operational\_crashfree\_hours

battery\_soc\_pct

battery\_soc\_pct\_drop

battery\_discharge\_duration

## App

Displaying fleet wide data on Device Vitals as cards in your dashboards.

### Connectivity

% Sync Success Rate

% Uptime

### Stability

% Stable hours

### Battery

Expected battery life

Observability for Embedded Devices

# Demo

# Built for **embedded** devices

## Power

- Has almost no measurable battery impact.
- Collects full system data to help you improve efficiency.

✔ **Works on your low power device**

## Connectivity

- Buffers data on device and sends when connectivity becomes available.
- Can chunk data into very small packet sizes ( $\geq 9$  bytes).

✔ **Works with any connectivity set-up**

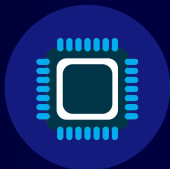
## Compute

- Does not add any additional performance load to system.
- Lightweight on device profile  
4.5kB FLASH, 1.5kB RAM

✔ **Works on highly constrained devices**

# Consistent across platforms

It works the same way, across all major embedded platforms



MCU



Android



Linux

# And there's **more**

## March 12th

Device Vitals

## March 13th

Better data, better  
dashboards, better drill down

## March 14th

Jira Integration



# Q&A