ServiceMeshCon

ServiceMeshCon

How (and Why) Google Manages Millions of Sidecars Without Breaking Everything

Sven Mawson (sven@google.com)

The Sidecar

ServiceMeshCon

Why a Sidecar?

- Prohibitive per-language expense
  - HTTP and RPC client and server libraries
  - Service Management (Networking, Security, Telemetry)
  - API Management
  - Other complex client libraries (e.g. Data Storage, Distributed File Systems, Messaging, Locking, ...)

Why a Sidecar?

- Ongoing per-language divergence
  - Default settings
  - Bugs (bug-for-bug compatibility)
  - Performance profiles

Why a Sidecar?

ServiceMeshCon

- Difficulties in rolling out fleet-wide changes
  - How many processes have bug X?
  - Rebuild, re-qualify, redeploy
  - What if it is a critical fix?

But why a *Sidecar*!

- Rejected Alternatives
  - Foreign Function Interfaces (e.g. SWIG)
  - DLLs
  - Remote Services

The Challenge

Roll out a sidecar to millions of processes, keep it up to date, and don't break anything.

Challenge Accepted!

Enabling Sidecars is Scary!

ServiceMeshCon

- Start with opt-in for those with something to gain
  - New languages (e.g. golang)
  - New functionality (e.g. Api Management)

Enabling Sidecars is Scary!

ServiceMeshCon

Enabling Sidecars is Scary!

- Make it fast, make it cheap
  - Multiply latency savings by # of hops
  - Multiply resource savings by # of sidecars
  - Never stop investing in performance improvements

Enabling Sidecars is Scary!

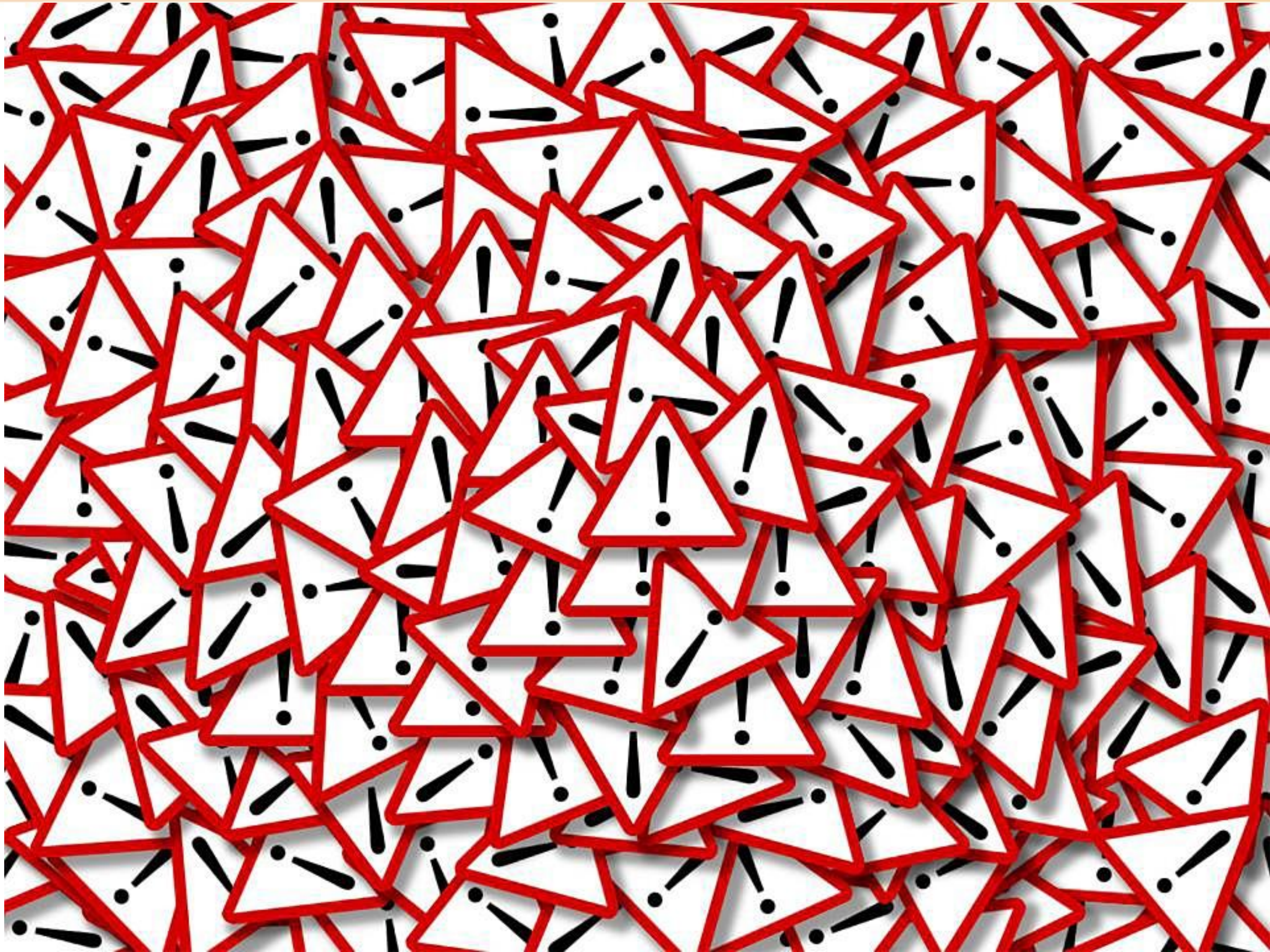
ServiceMeshCon

- Proactively identify those most affected
  - Anyone close to a resource limit already
  - Anyone with a business-critical job

Enabling Sidecars is Scary!

ServiceMeshCon

Warn
Your
Users!



Enabling Sidecars is Scary!

- Warn them!
    - And warn them again. And again.
    - And again.

Enabling Sidecars is Scary!

ServiceMeshCon

Enabling Sidecars is Scary!

ServiceMeshCon

- Enable it by default slooooooowly
  - Cluster by cluster, job by job
  - Make opt-out easy, follow up later

What About Upgrades?

Upgrading is also Scary!

ServiceMeshCon

- Two tracks: dev and stable
    - dev is weekly, stable is monthly

Upgrading is also Scary!

ServiceMeshCon

Keep
It
Simple!

Upgrading is also Scary!

ServiceMeshCon

- Two modes: automatic and bound
  - Automatic happens on restarts (~monthly)
  - Bound mode sets version = latest when built
  - Bound mode requires agreement to:
    - Update regularly (at least every 2 weeks)
    - Update quickly for critical fixes

Upgrading is also Scary!

ServiceMeshCon

Upgrading is also Scary!

ServiceMeshCon

- Roll out updates slowly!
  - Qualify new releases with critical users
  - Start with canaries (automatic + bound)
  - Slowly ramp to 100%
  - Take your time, especially early on!

Upgrading is also Scary!

ServiceMeshCon

Don't
Make
Them
Grumpy

Upgrading is also Scary!

- Make rollback self-serve
  - Allow users to roll back to a specific supported version
  - Keep a dashboard of how many users rolled back
  - Be proactive and fix their issues ASAP
  - Roll the whole thing back if too many users broke

OK, So What Did We Learn?

Lessons Learned

ServiceMeshCon

- Invest in debugging tools
  - Whose fault is it?
  - How often is it happening?

Lessons Learned

ServiceMeshCon

- Application <-> Sidecar comms need to be rock solid
  - Tried an experimental shared memory channel
  - Ended up with Unix Domain Socket: slower but much more reliable

Lessons Learned

ServiceMeshCon

- You're going to take a performance hit
  - Batch chatty connections
  - Keep investing in performance!

Lessons Learned

ServiceMeshCon

- Avoid changes in default behavior
  - Dynamically load functionality on demand
  - Guard new functionality with opt-in and opt-out
  - Be especially careful of newly opened connections!

Lessons Learned

ServiceMeshCon

- Don't get fancy
  - Share fate between application and sidecar
  - No hot reloading, pick up new versions on restart
  - Avoid one-offs
  - Make it easy to run the sidecar during development
  - Make it easy to run the sidecar in tests

Lessons Learned

It is Totally Worth It!

Questions?