



#SREcon EMEA
August 31, 2018

Tradeoffs in Resiliency: Managing the burden of data recoverability

Kristina Bennett (@kilobitten)
Site Reliability Engineering, Google

Google Cloud

In this talk...

- What is recoverability?
- Why is it difficult?
- How can I make sure I have some?
- How do I know how much I need?

Hi!

Kristina Bennett, @kilobitten

At Google since 2009

SRE since 2013

Now: Cloud Customer Reliability Engineering



@kilobitten, #SREcon EMEA

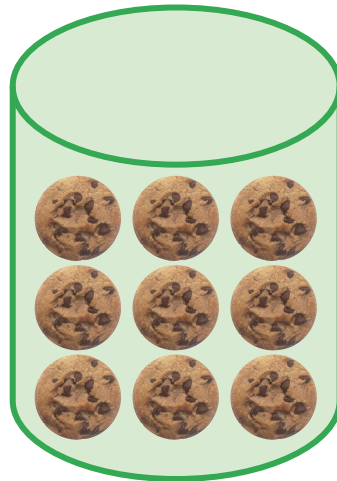


The recoverability game

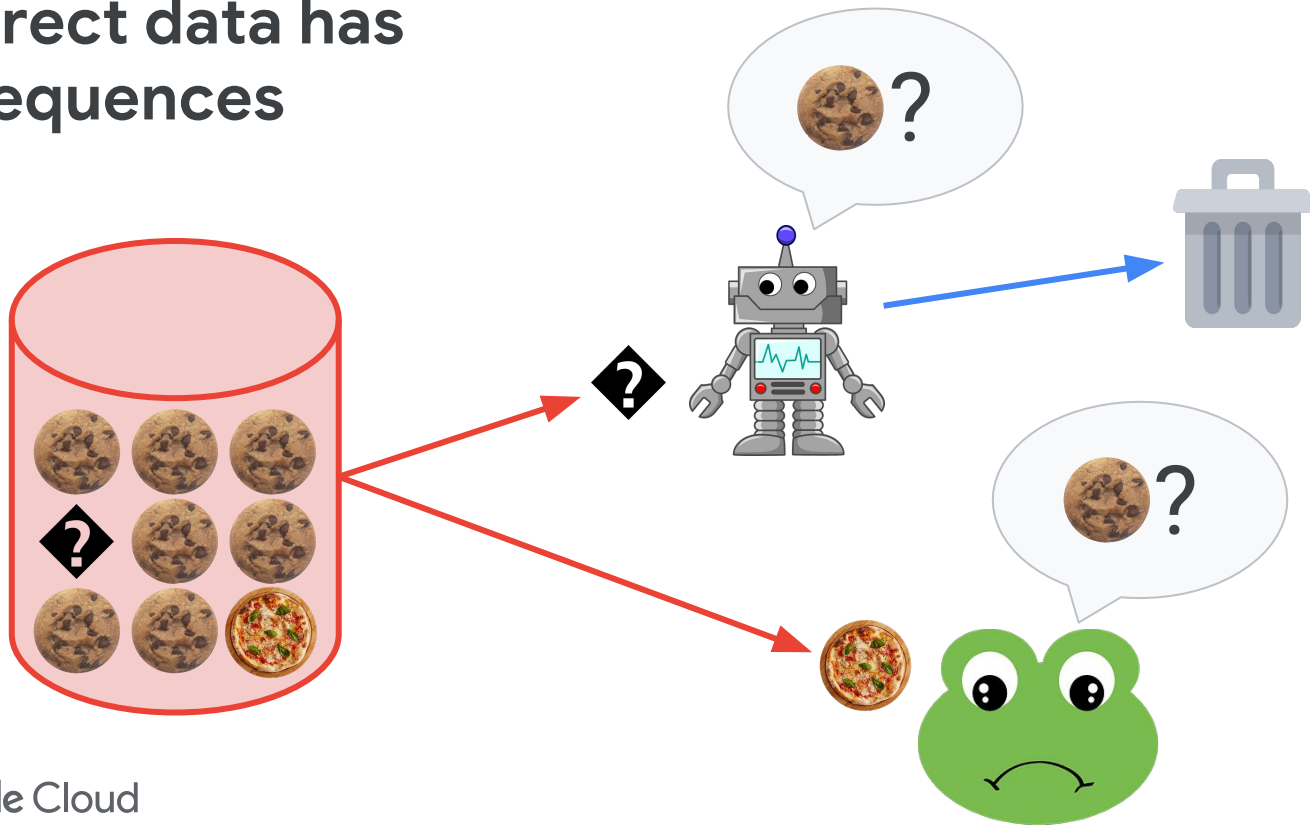


**You store
persistent data**

Snack DB



Incorrect data has consequences



Advanced mode: Large scale data



?



Recoverability is hard



@kilobitten, #SREcon EMEA

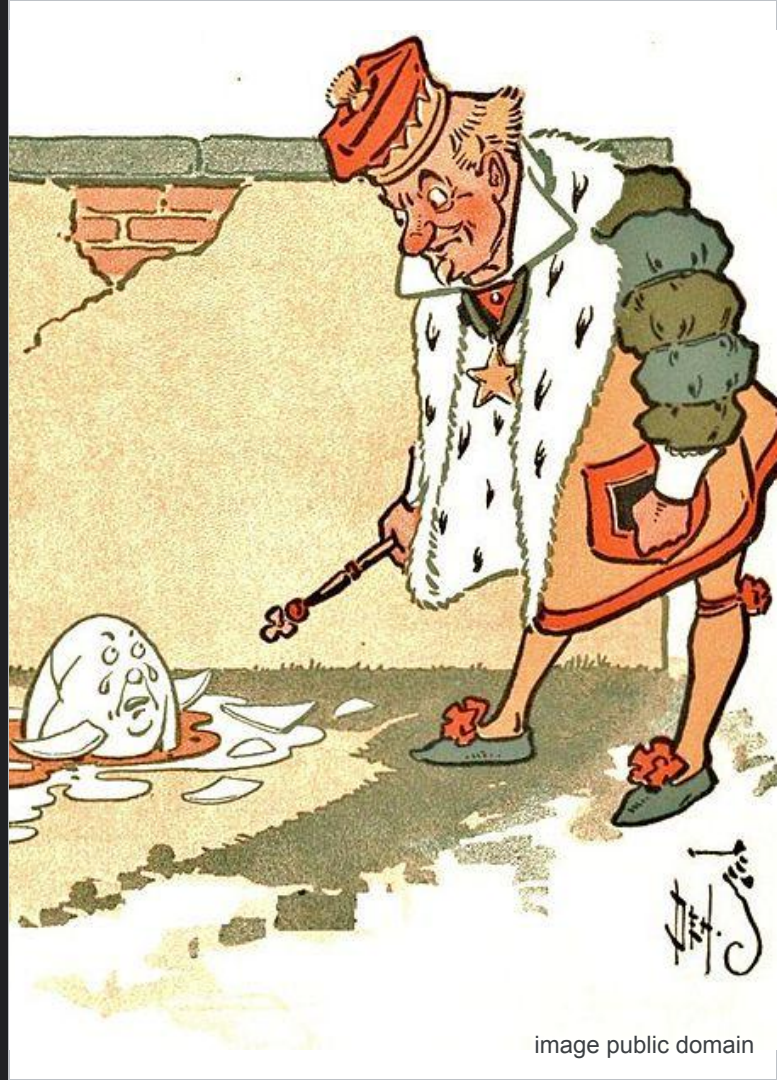
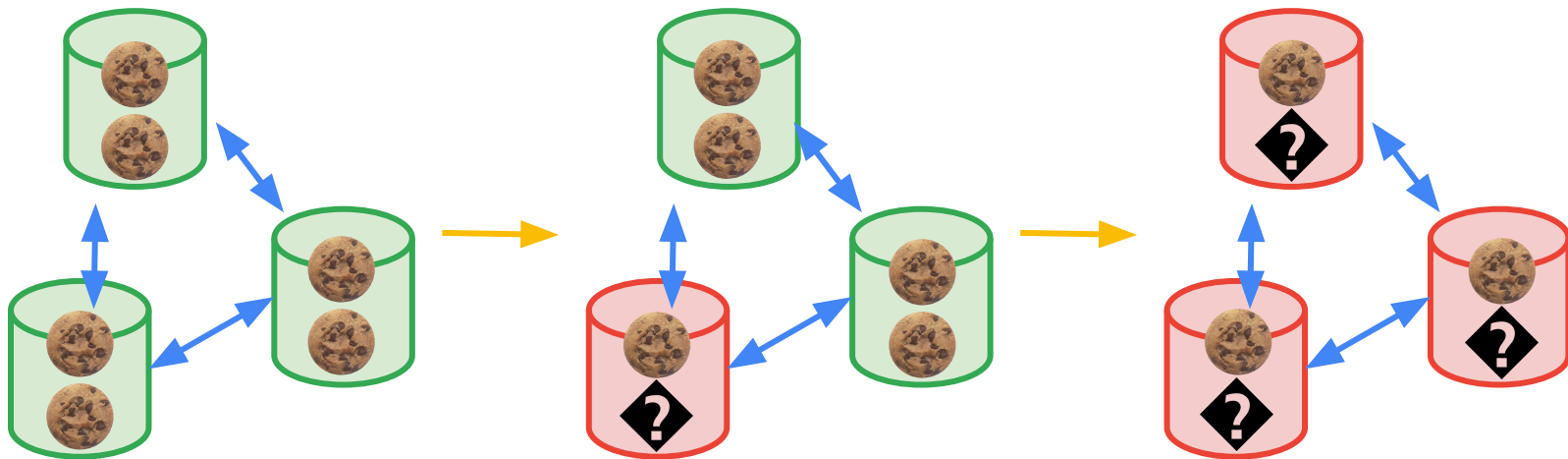


image public domain

Redundancy is not durability

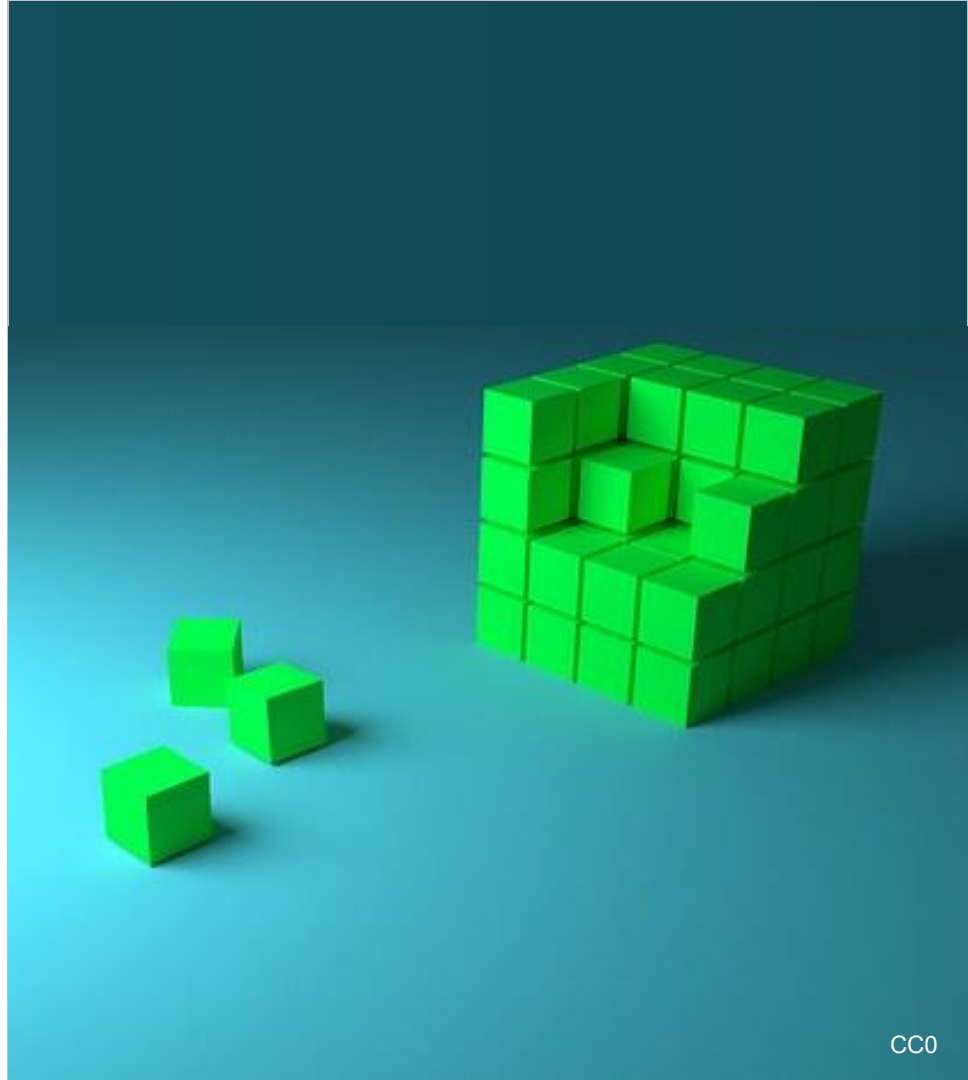
At least, not on its own.

Corrupted writes and deletes replicate beautifully.



What *can* provide durability then?

- Rebuildable data
- Replayable changes
- Backups



A backup is not a restore

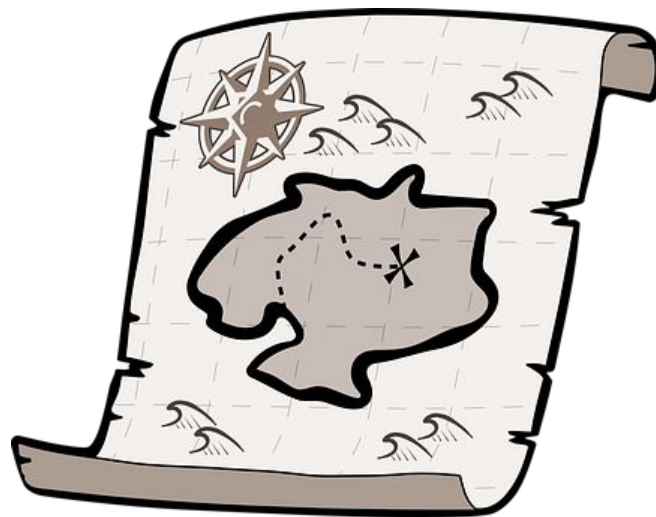
Have you actually
retrieved a backup?



Where are the backups?



You know you stored them somewhere.



How long will it take you to track them down?

Was the backup process working properly?

Hey...
where are the backups?



@kilobitten, #SREcon EMEA



[CC-BY Scott Meyers. "empty storage unit"](#), image cropped

Which backup is the one you need?

Are they meaningfully
named or indexed?

Do you know precisely
how far back you need to
look?



@kilobitten, #SREcon EMEA



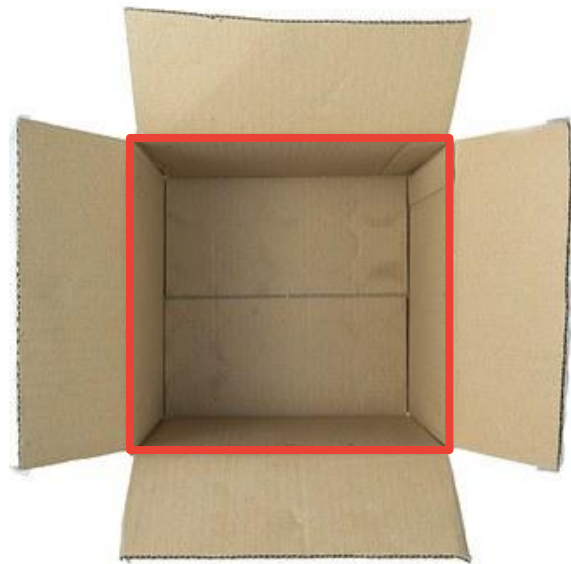
Are you sure the backup contains what you expected?



You expected this....



but the config was wrong...



or possibly invalid...

Do you have room to unpack the backup?



The backup was probably
compressed/remote...



so you'll need to stage it somewhere.



A restore is not a recovery



@kilobitten, #SREcon EMEA



(c) Craig Bennett, used with permission.

A snapshot lives in the past



@kilobitten, #SREcon EMEA



CC0

Most recoveries need to patch in partial data



**Newer data may
need to be
re-merged**



**Dependencies will
need updates or
repairs**



Recovery includes letting the dust settle



@kilobitten, #SREcon EMEA



CC0

How can you prepare?

Step 0: Know your storage



@kilobitten, #SREcon EMEA



What are you storing?

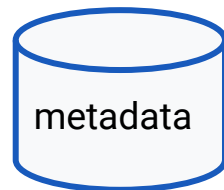
"Our primary data is our cat gif repository."



And the metadata index.

What are you storing?

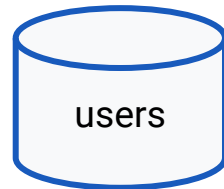
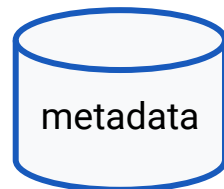
"Our primary data is our cat gif repository and the metadata index."



And the user account database.

What are you storing?

"Our primary data is our cat gif repository, the metadata index, and the user account database."



And the "Most popular gif" scores.

What are you storing?

"This document is a full description of our primary datastores."



random_init_
file_that_we_
need_or_
everything_
crashes.init



metadata

users

user
comments

popularity
scores

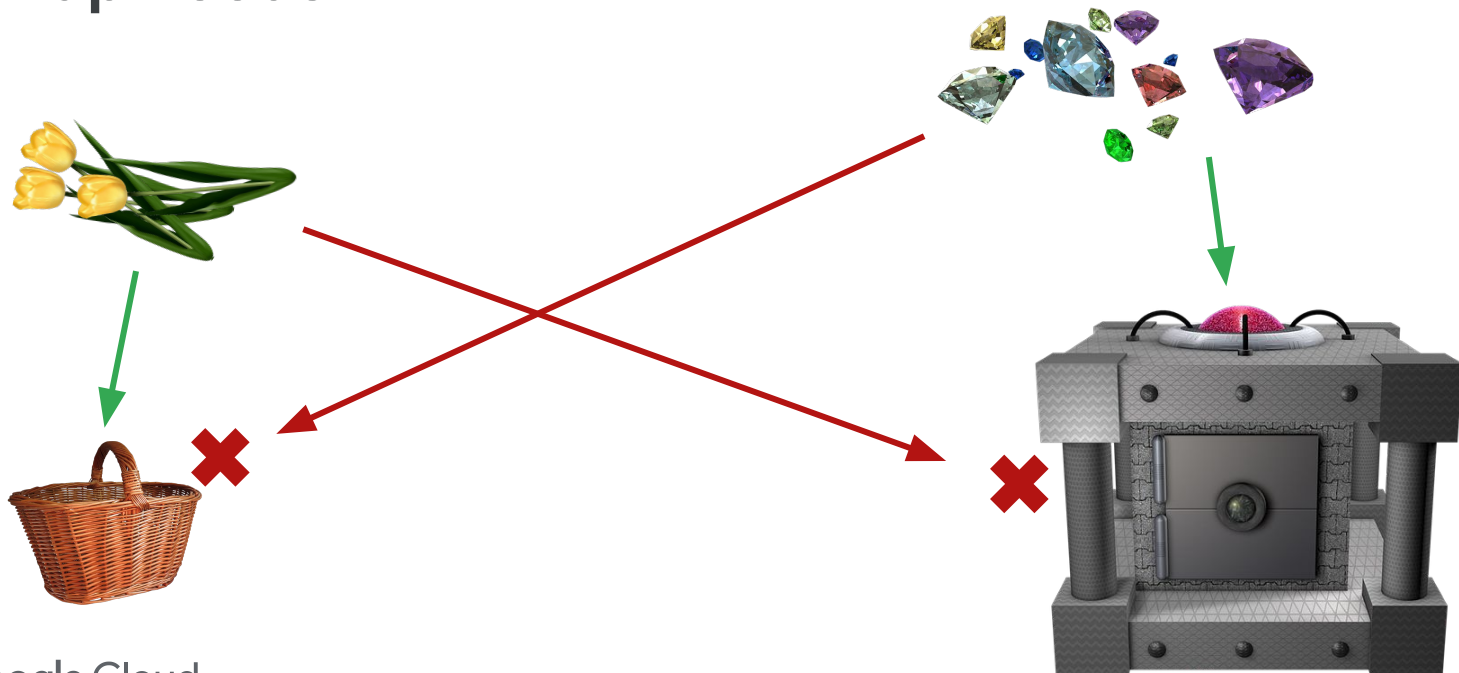
Know your storage

- What do you store?
- Where and how do you store it?
- Where does it come from?
- How long do you keep it?
 - Why? What dictates the min/max retention limits?

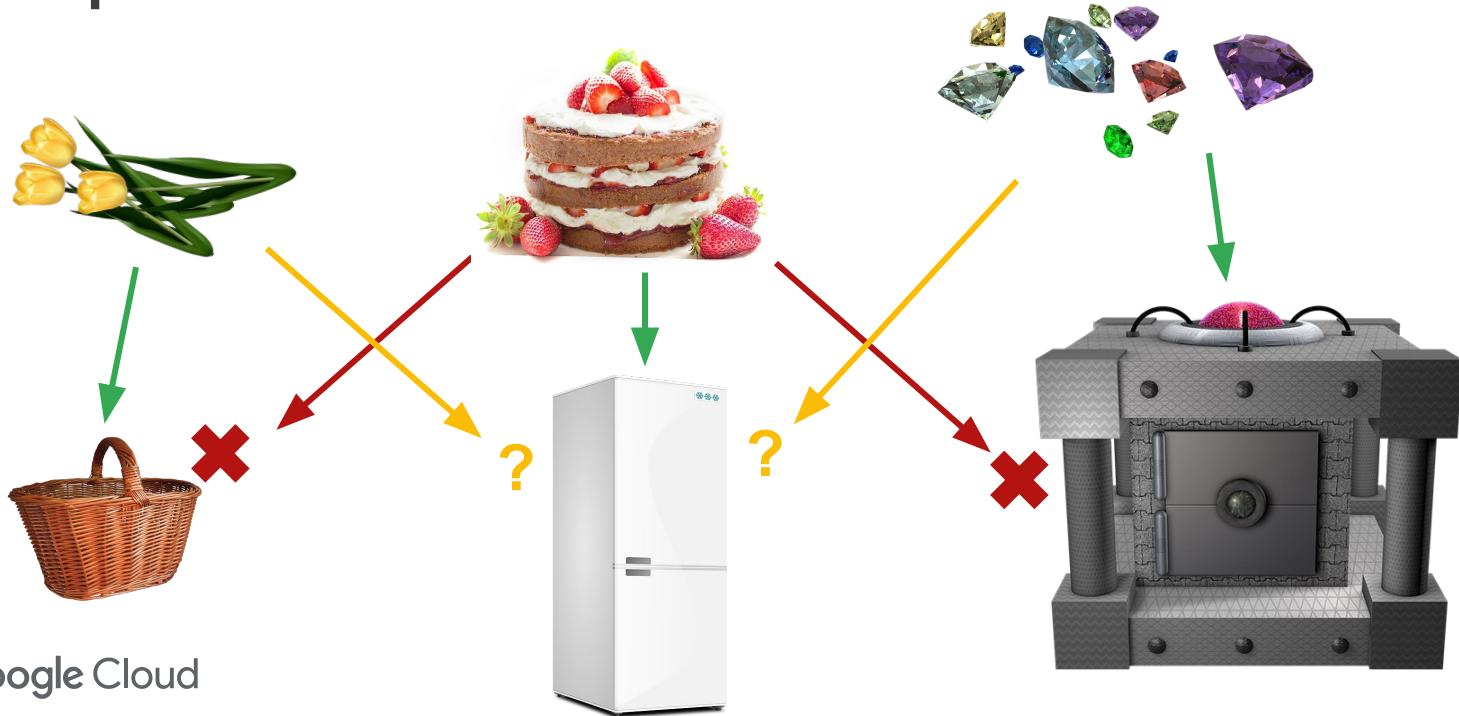
Consider your recovery options

- More than just backups
 - *Can you rebuild or replace?*
- Back up strategically
 - *Retention tiers can scale backup coverage and conserve resources*
- You're going to need some tooling

Different datastores, different backup needs



Different datastores, different backup needs



Prepare Validations

- What comprises "valid"?
 - *How will you know if a recovery is successful?*
- Bonus: now you have a detection tool!
 - *Pro-active consistency checking: finding problems before they escalate.*

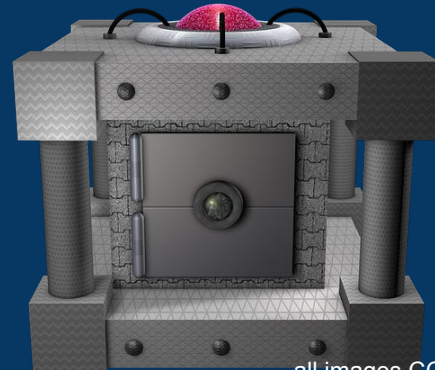


Recovery plan

- Have one!
- Document it
- Test it
- Automate it



Resiliency Tradeoffs



Advice so far, recap:

1. **Catalog** your storage
2. Design a recovery **plan**
3. **Prepare** the chosen recovery options
4. **Test** the plan regularly
5. **Automate** recovery tasks



**That's... a *lot* of
work.**



Fit for purpose!



@kilobitten, #SREcon EMEA



CC0

What guides SRE choices?

**Business needs +
Technical constraints**



SLOs & Error Budgets



Risk assessment



Business Requirements + Technical Constraints

How resilient is the service to losses?

What effect will a data incident have?

SLOs & Error Budgets



- Product management & SRE define a **reliability target (SLO)**.
- (100% - reliability target) is a “budget of unreliability”, i.e., the **error budget**.
- Monitoring measures **actual performance**.
- Control loop for data-driven decisions!



Business Requirements + Technical Constraints

How resilient is the service to losses?

What effect will a data incident have?

SLOs & Error Budgets

~~How resilient is the service to losses?~~

- *What is the service's durability error budget?*
- *What is the service's availability error budget?*

~~What effect will a data incident have?~~

- *What is the availability impact?*
- *Over what % of the service?*

Risk Assessment

~~How resilient is the service to losses?~~

→ *What is the service's durability error budget?*

→ *What is the service's availability error budget?*

~~What effect will a data incident have?~~

→ *What is the availability impact?*

→ *Over what % of the service?*



How much error budget will an incident burn?



Explore improvements

Risk Name	ETTD (mins)	ETTR (mins)	% Users	ETBF	Bad mins/year	Δ	% change
Operator accidentally deletes database; restore from backup required	5	480→360	100	1460	121→91	30	25
Bug in new release corrupts 1% of write-backs	1440	600→120	2	90	166→127	39	24
Timestamp bug in new release triggers widespread premature garbage collection	90	720→240	95	1095	257→105	152	59
Bad storage configuration change truncates the largest 5% of entries	240	660→180	15	1460	135→63	72	53

Some Risk vs. Cost Tradeoffs

Need	Risk	Cost
Restore and Recovery Tools	Lack of needed tooling	Development
Backup Retention	Time limit on recovery	Storage management
Backup Locality	Longer time to recovery	Resource contention and expense
Recovery Independence (Flipside of Locality!)	Recovery resources aren't independent of serving	Reduced locality
Recovery Precision	Limits on accuracy of point-in-time recovery	Increased recovery data creation/retention

Wrap up

Recoverability is hard

- Redundancy != durability
- Restores > backups
- Recoveries > restores

You can be prepared

1. **Catalog** your storage
2. Design a recovery **plan**
3. **Prepare** the chosen recovery options
4. **Test** the plan regularly
5. **Automate** recovery tasks

Make data-driven decisions

SLOs...

- determine error budgets, which provide context for risk assessment
- provide constraints for risk vs. cost tradeoffs

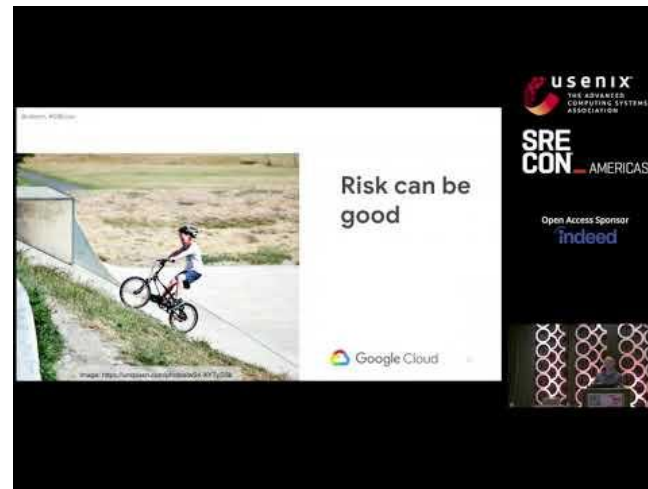
Video Resources



Liz Fong-Jones at CodeAsCraft at Etsy -
Adopting SRE and Error Budgets

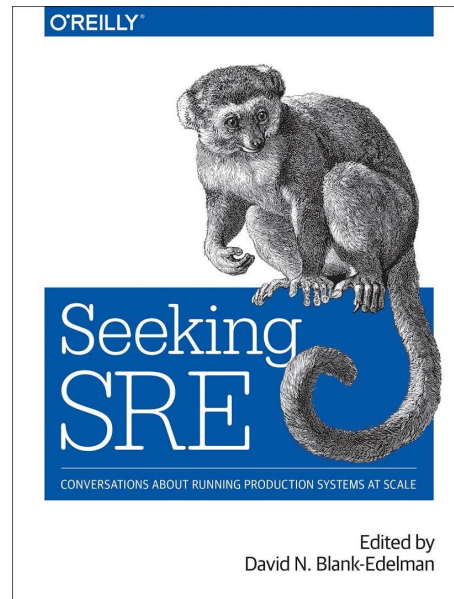
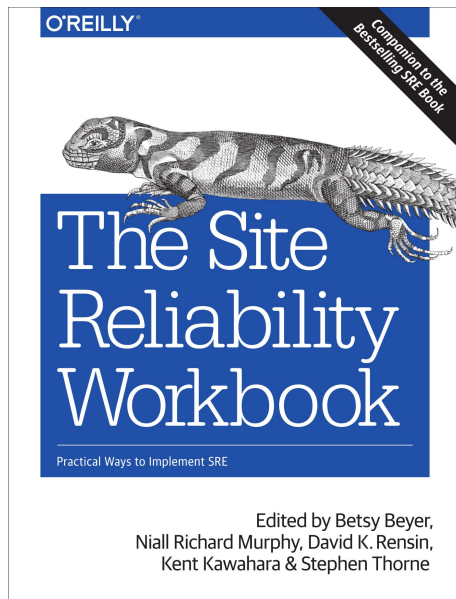
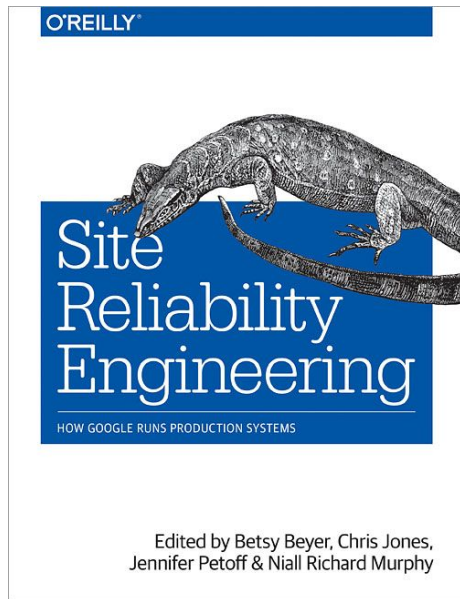


@kilobitten, #SREcon EMEA



Matt Brown at SREcon18 Americas -
Know Thy Enemy: How to Prioritize and
Communicate Risks

SRE resources



@kilobitten, #SREcon EMEA

Cover images used with permission.
These books can be found on shop.oreilly.com.

Thank you to the awesome colleagues who made this talk possible!

- Data Integrity Team
- Customer Reliability Engineering
- Special thanks to Liz Fong-Jones

Questions?

