THE UNIVERSITY OF QUEENSLAND
AUSTRALIA
CREATE CHANGE

UQ's Long Term Storage Journey.

Co-design, risk and value.

Jake Carroll, Chief Technology Officer, Research Computing Centre, The University of Queensland, Australia.

jake.carroll@uq.edu.au

Story time...

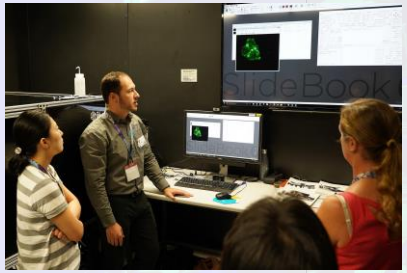
- UQ RCC is responsible for the design, implementation, management, maintenance and support of the infrastructure that houses ~93% by volume of UQ's Scientific Research Data.
- We set out to build new infrastructure for this task, ~3 years ago.
- This is the abridged story.

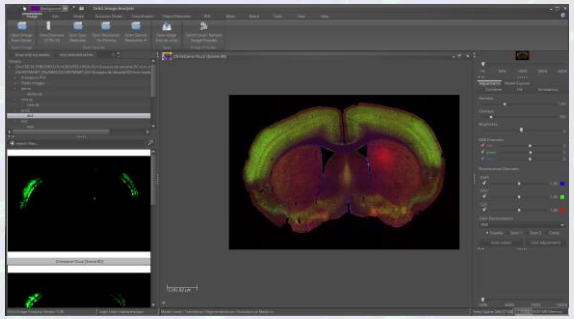
Context

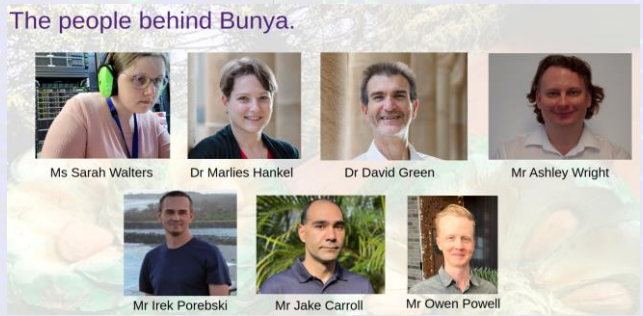
UQ is:

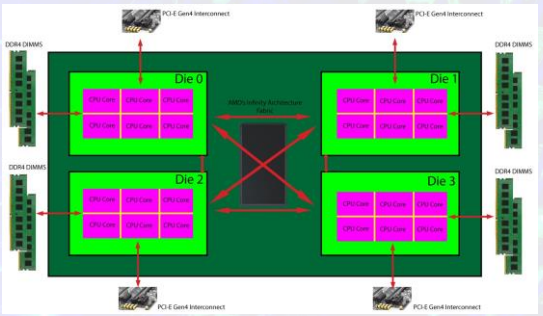
- Six major faculties
- Eight institutes
- Fifteen sites
- ~55,400 students (2022)
- ~7,410 full time staff (2022)
  - 4000 researchers
- ~25,000 endpoints
- Tier2 supercomputer: more than 10,000 CPU cores, cutting edge GPUs.
- ~100 PB of research data storage under management.

UQ Research Computing Centre

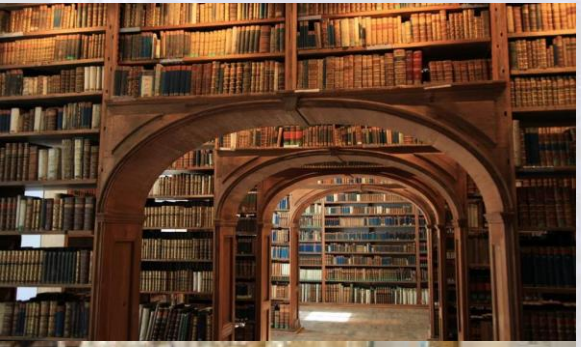








Scientific research data storage infrastructure for preservation and archiving.



A special case for data storage systems with focus and strength in certain attributes such as low TCO, durability, retention and expandability.

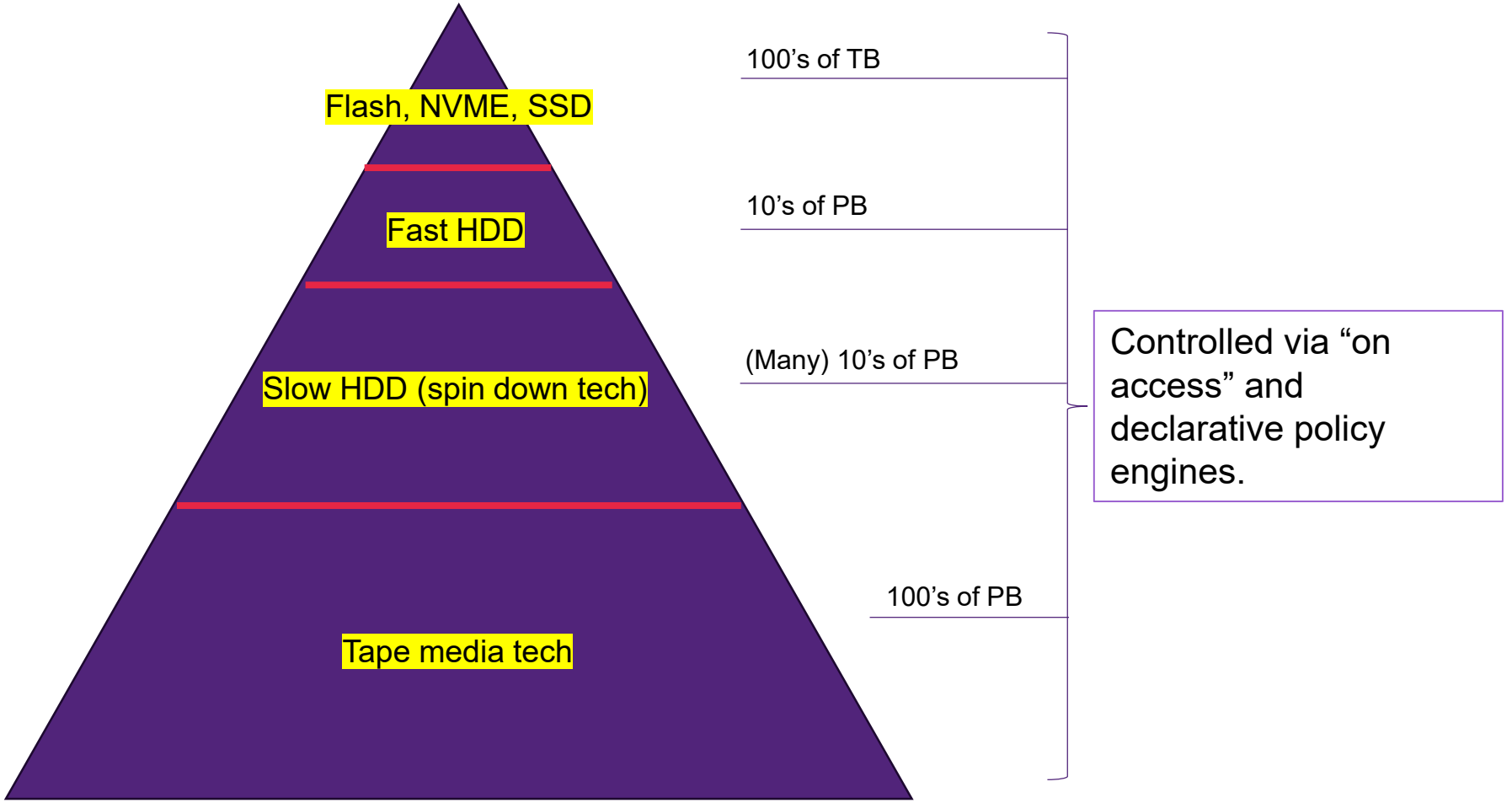
These attributes may be artefacts of a generalised Research Data Reference Architecture with the following features:

- Resilience, Discoverability, Governability, Manageability, Accessibility, Scalability, Versatility, Security.[1]

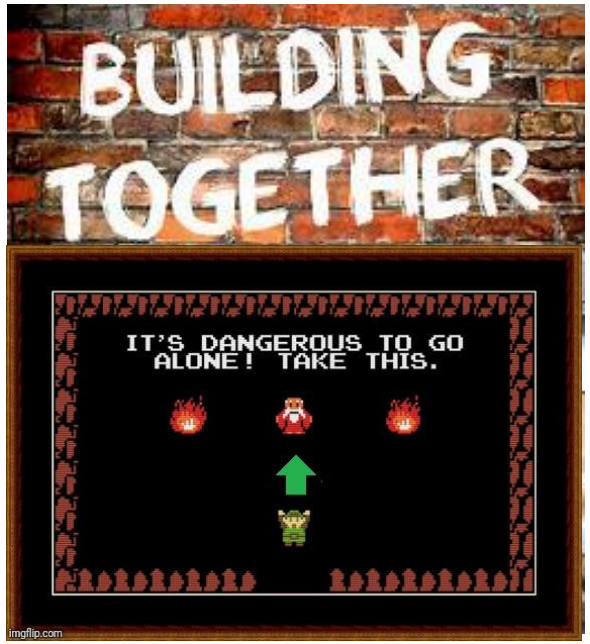
[1]Abramson, D., Betbeder-Matibet, L., Bird, S., Carroll, J., Francis, R., Goscinski, W., Soo, A., Swan, G., *Why we need a Reference Architecture for Research Data.* 2023

The storage hierarchy (*tiers*)

"HSM" : Hierarchical Storage Management. This is how we keep data preserved, retained, available and affordable on the UQ "Q" Collections.

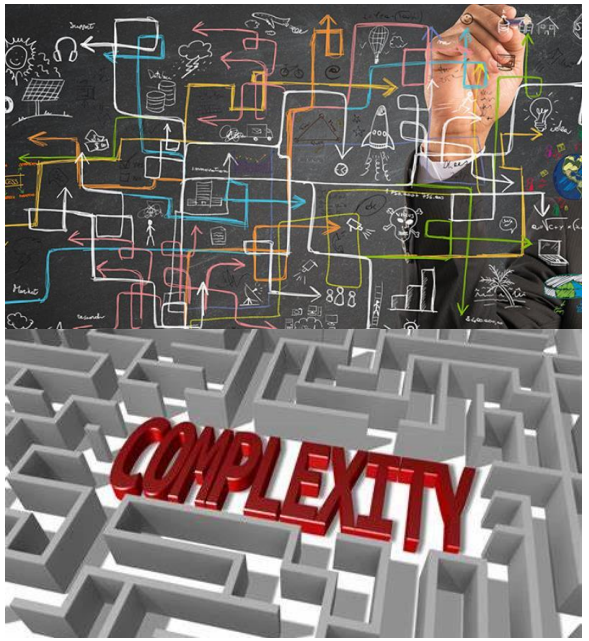


BLUF or TL;DR



You don't go on this trip alone. *Partnership is needed.* Learn how shared risk models work in high end technology procurement.

Buckle up. Leadership class systems to get world first capabilities and deliver new value **ARE** going to be hard.



Significant scale and flexibility can confer significant complexity but can create administrative burden.

Simplicity can confer ease of administration, but can also infer rigidity, cost inefficiency and less capability.

ur·gen·cy

[ˈəːdʒ(ə)nsi]

NOUN

1. importance requiring swift action:
   "the discovery of the ozone hole gave urgency to the issue of CFCs" · "these are the grave urgencies facing us"

   SIMILAR: importance | top priority | imperativeness | weight | weightiness

2. an earnest and persistent quality; insistence:
   "Emilia heard the urgency in his voice"

   SIMILAR: insistence | persistence | determination | resolution | tenacity

UQ didn't have any room to "*wait it out**"
4.6 billion files, hitting all the limits.
DMF6 was close to not being fit for purpose, anymore.

*A bigger question: Wait it out for ***what*** exactly?

What did we ask the market for?

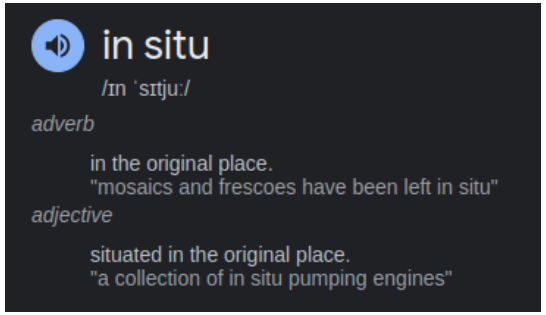
IBM Storage Scale DMAPI, Light Weight Kernel Events → *Space managed by "the Solution".*

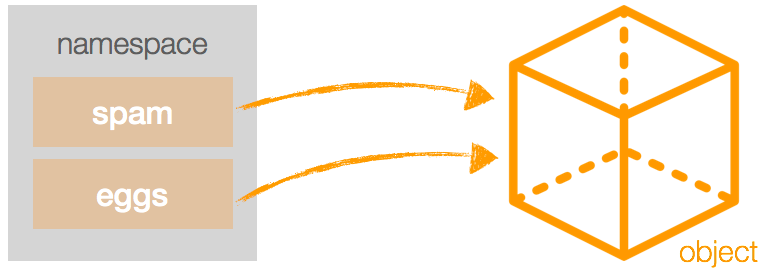
Storage Scale AFM Cache, filesets

Policy Engine controls all data management


```
objectset
keep_three_versions:
object.path like
'/user/path*' and ...
```

cxfs





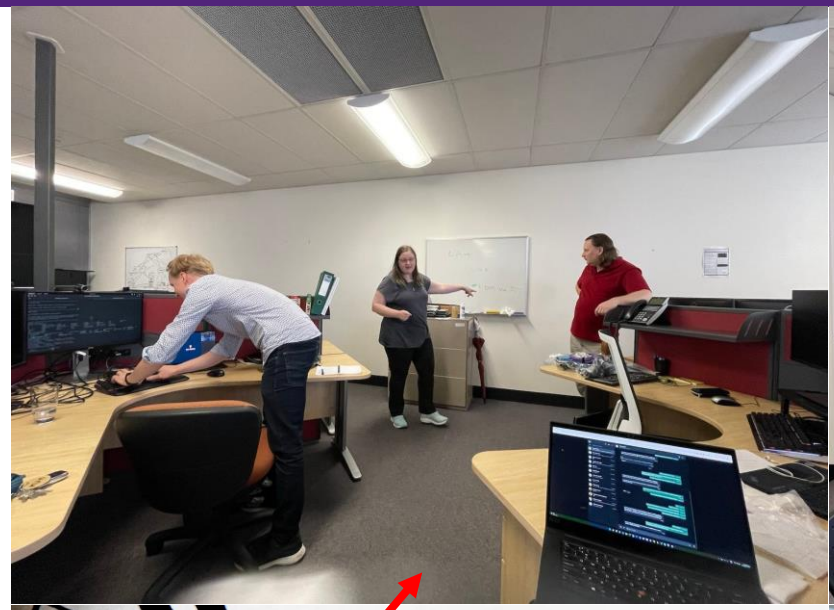

```
"query" : "object.path like
'/gpfs/UQ04/pool0401/Q0117/Q0117*'
and object.version in [:-1]",
```

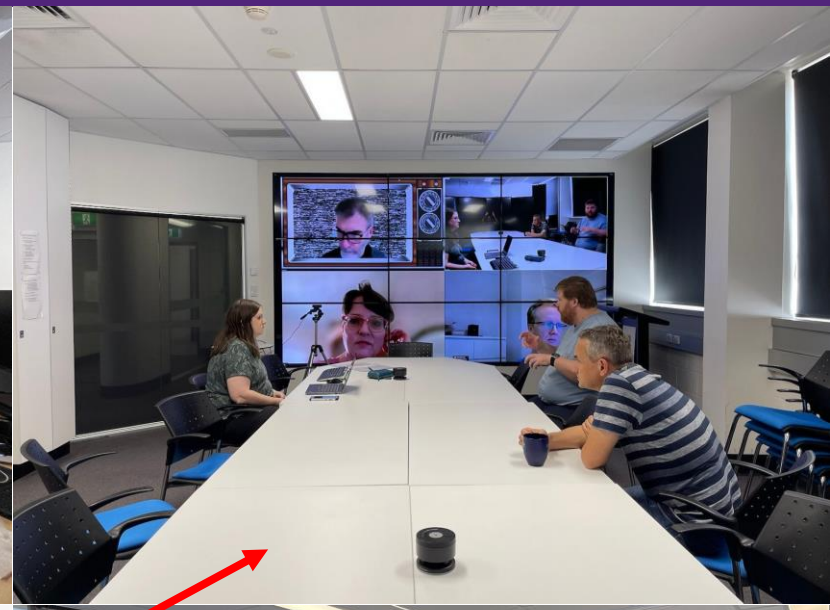
A long procurement process to find a replacement technology...

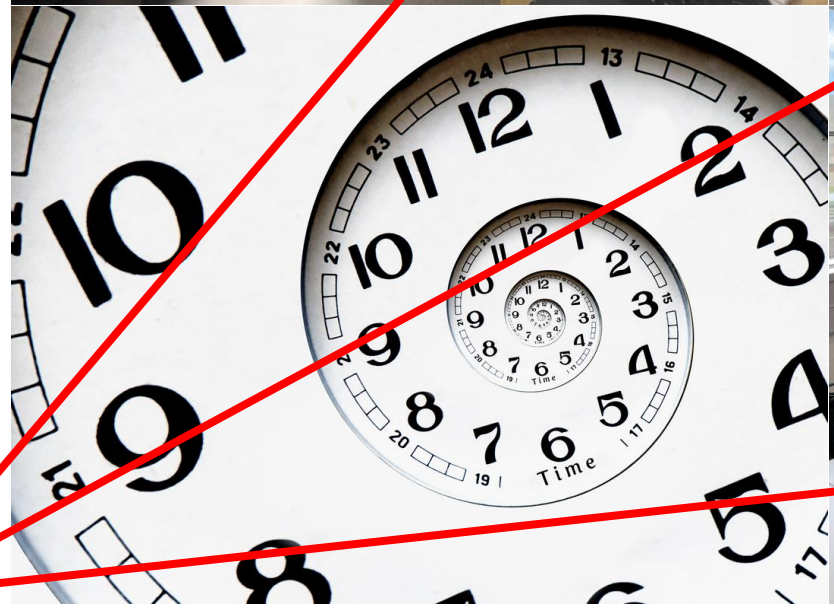
Results were **frustrating**:

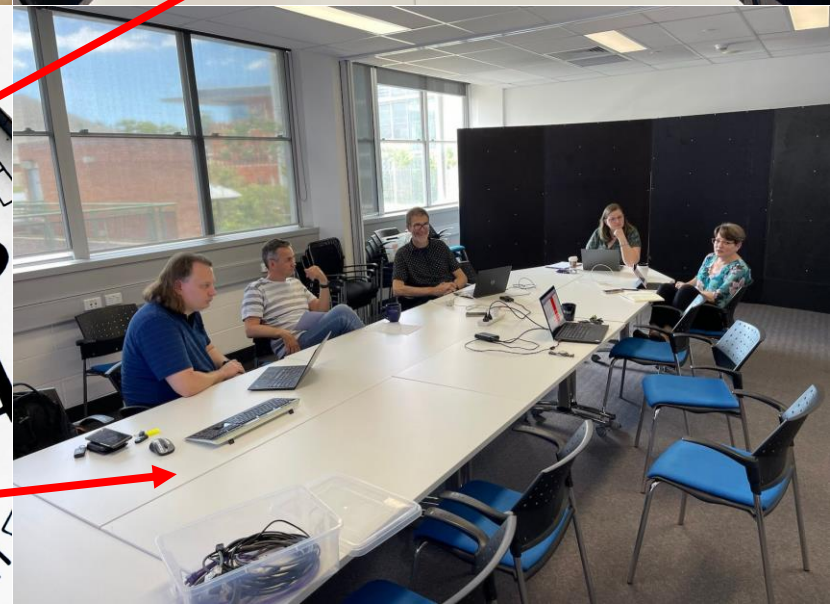
- Market responded poorly.
- Not enough dynamism.
- Of the next-gen products that were out there, they were too far off to be of use.
- Nothing screamed *"this is a 10+ year proposition"*
- Someone even told me I could put it all in the public cloud, it was *"easy"* and *"don't worry about the cost right now. That's the best thing about OpEx consumption models!"*. Good one, champ!

After a long time, we settled on HPE's DMF 7









NB: Some people that did hard procurement work…

Comments from peers, industry and colleagues in 2020...

- *"We heard the product barely functions! Are you sure this is ok?"*
- *"The product I knew and loved is dead!..."*
- *"It can't even archive to tape!"*
- *"I heard it's a database pretending to be a filesystem!*
- *"There is no migration path. It's a fresh start!"*
- *"It doesn't have an API!"*


RISK


WON'T SOMEBODY PLEASE
THINK OF THE CHILDREN!

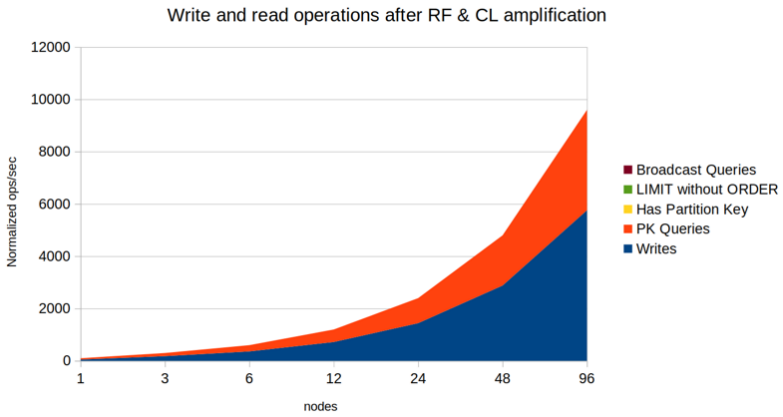
As usual, the truth was somewhere in the middle.

Day 0 pain points

- We were the first site in the world to jump from DMF6 to DMF7.
  - Migration from DMF6 BFID to DMF7 Cassandra NoSQL wasn't easy.
- We still managed to get up and running in our change window.
- Query engine was unoptimised for our scale.
  - When you're bordering on 5 billion objects and HPE labs had only tested contrived examples, it was always going to be odd.
- Nobody understood the hardware scaling initially. It was wrong and needed remediation.
- Quality of life for administrators: **hurting. Not fun. 3/10.**
- Quality of life for users: **bumps and inconsistent. 5/10.**

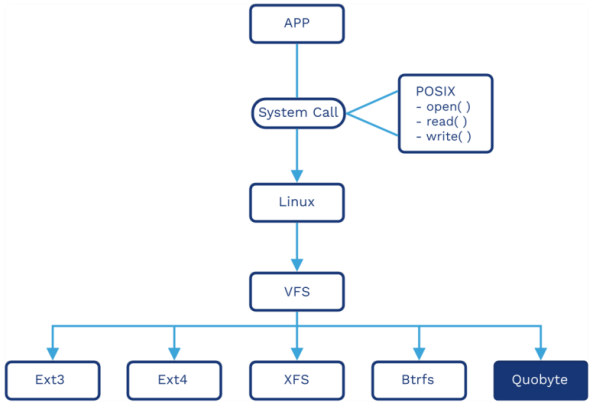
Reality:

functional



Scalability issues out of box

BURIAL vs
CREMATION
Rest in pieces, DMF 6



Was this POSIX anymore? Yes?
The front bit people “see” at least?”

In place migration: yes.

Day ~100 pain points

- Users are happier but promise of query-able filesystem isn't real yet, in practicality.
  - *An example:* If I wanted to query my entire ZeroWatt Storage consumption (pushing 35PB, alone) and “trim” it daily so deleted or stale data is removed: could I do that? **No**, not tenably.
  - If I wanted to run a simultaneous copy policy to replicate (old) objects in two tape libraries with 20 * TS1160 tape drives (*that’s a lot of IO, btw...*), could I? **No**, not tenably.
  - Filesystem event filters, scanners, memory consumers, table structures all had optimisation issues. A lot of all hands on deck, just to keep it all on the level.
- "*Hey Owen? Can you restart the data manager agents again please?*" x 100,000,000
- Assistive tools for admins were slim or non-existent.
- Snapshots are hard. Migration from CXFS filesystems is proving really tricky. Tape migrations? Not a reality, yet.
- Quality of life for administrators: **Stings a bit still. 6/10.**
- Quality of life for users: **better but corner cases still hurt experience. 7/10.**

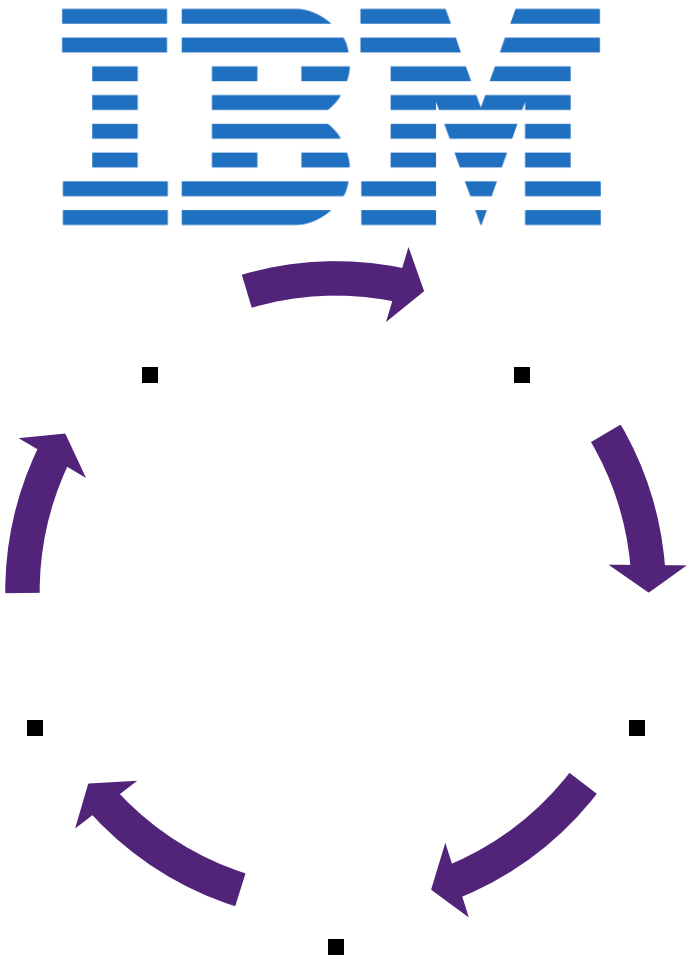
Day ~1000 pain points

- We need more reporting functionality "baked in". Currently reporting is a bit difficult and we need to re-tool to do usage stats in a more uniform and simple way for our brothers and sisters upstream to digest (UQ Research Systems team/RDM crew).
- Some more user-land tools, perhaps? *[Maybe…maybe not needed?]*
- An "all in one" dashboard for a global view of the platform. `Check_mk` is good but not what I'm looking for at the executive summary level.
- Quality of life for administrators: **Feeling better (ish). 7.5/10.**
- Quality of life for users: **We reached HSM "transparency". 9/10.**

Day ~*now* pain points

- It still feels like we are learning how to drive.
- Predictability of a NoSQL environment needs some more work.
- "forward recall" type (very) advanced HSM functions yet to come.
- Knowing what to tune, how to tune it.
- Tooling for very advanced migrations is all there, but not done without HPE's advisory and help. *"These are not the undocumented commands you are looking for"*
- Quality of life for administrators status: **Better vibes. 8.5/10.**
- Quality of life for users: **The data fabric experience we wanted for our users. Cake, ate it too. 9.5/10.**

Collaborative work: Knitting a data fabric, together...

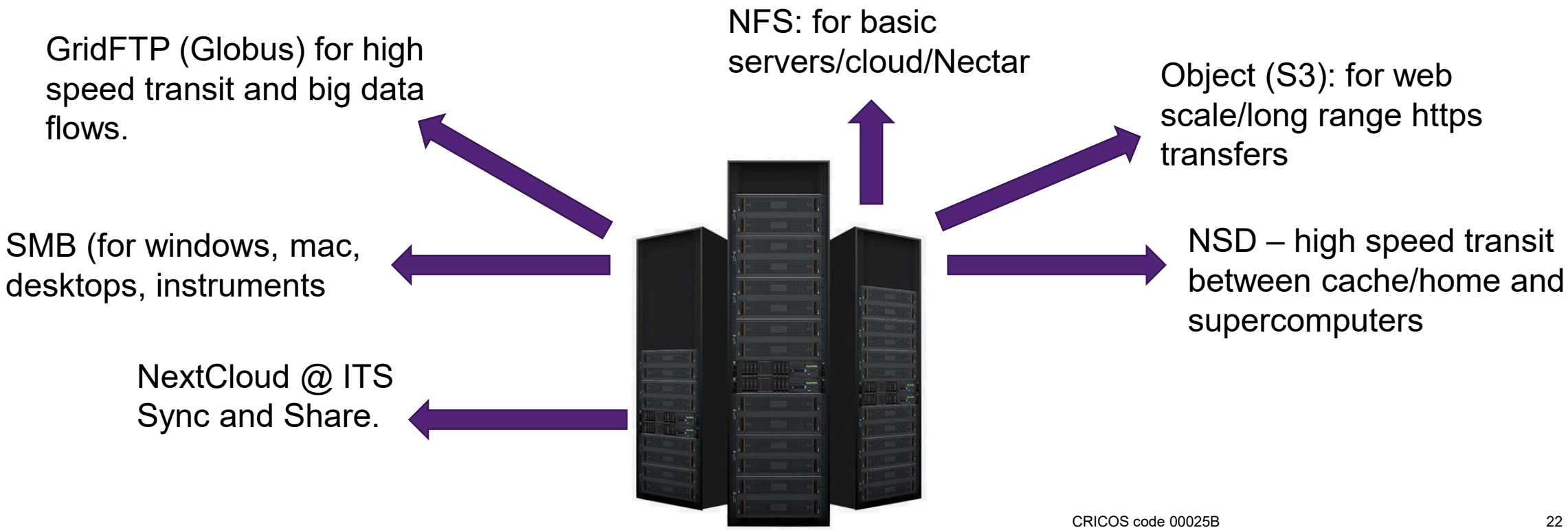


Hewlett Packard Enterprise

The University Of Queensland
AUSTRALIA

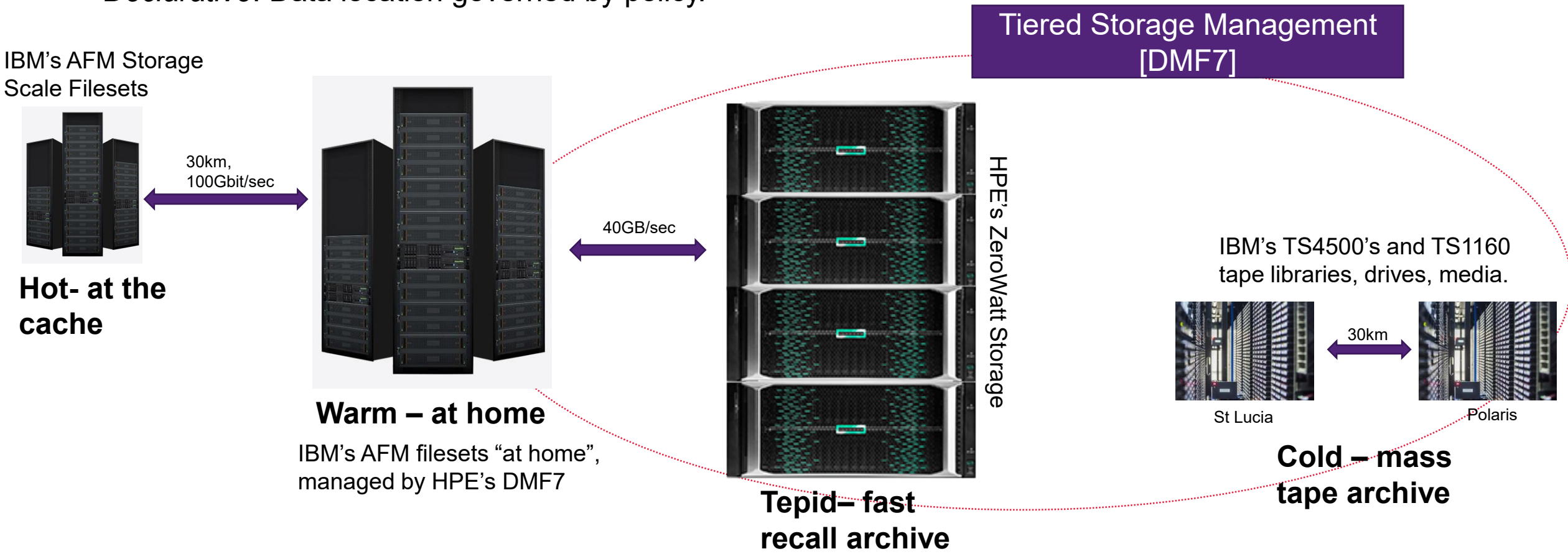
Offer data in a way that suits the use case. **Protocols.**

- Recognise that different users, workloads and communities have different ways they present their data.
- Use technology that can present a view of the data in many forms, from one namespace.



Data where it needs to be, when it needs to be there.

- Using tiering technology and the concept of caches so that data is where it needs to be *on-access* and not taking up space on expensive media, when it is cold and infrequently used.
- *Declarative*. Data location governed by policy.



On the other side of the AFM home, we've natively integrated HPE's DMF7 to space manage Storage Scale in our ESS.

ESS 3500+5000
Storage Scale
Filesystems

```
/UQ00
/UQ01
/UQ0..
/UQ08
```

Light weight events via Ganesha
interface processed by DMF7
to events_filter for DMAPI processing

dmf7-gpfs-dm1
dmf7-gpfs-dm2

200G HDR fabric

MESOS

Apache Zookeeper

Cassandra Object Scale db,
HDR200 connected event
processing and reflection table
generation cluster

Spark

MARATHON

A more holistic picture of the new architecture shape…

Research Data Manager

Q collection
Request from User

Cache @ IMB

Cache @ QBI

Cache @ ITS

UQ OneNet 100G Core

25km of highway

Polaris 100G Edge

DMF7 engine
- Policy
- Placement
- Preservation
- Data movement
- Integrity

cassandra

/afm
DMAPI space managed

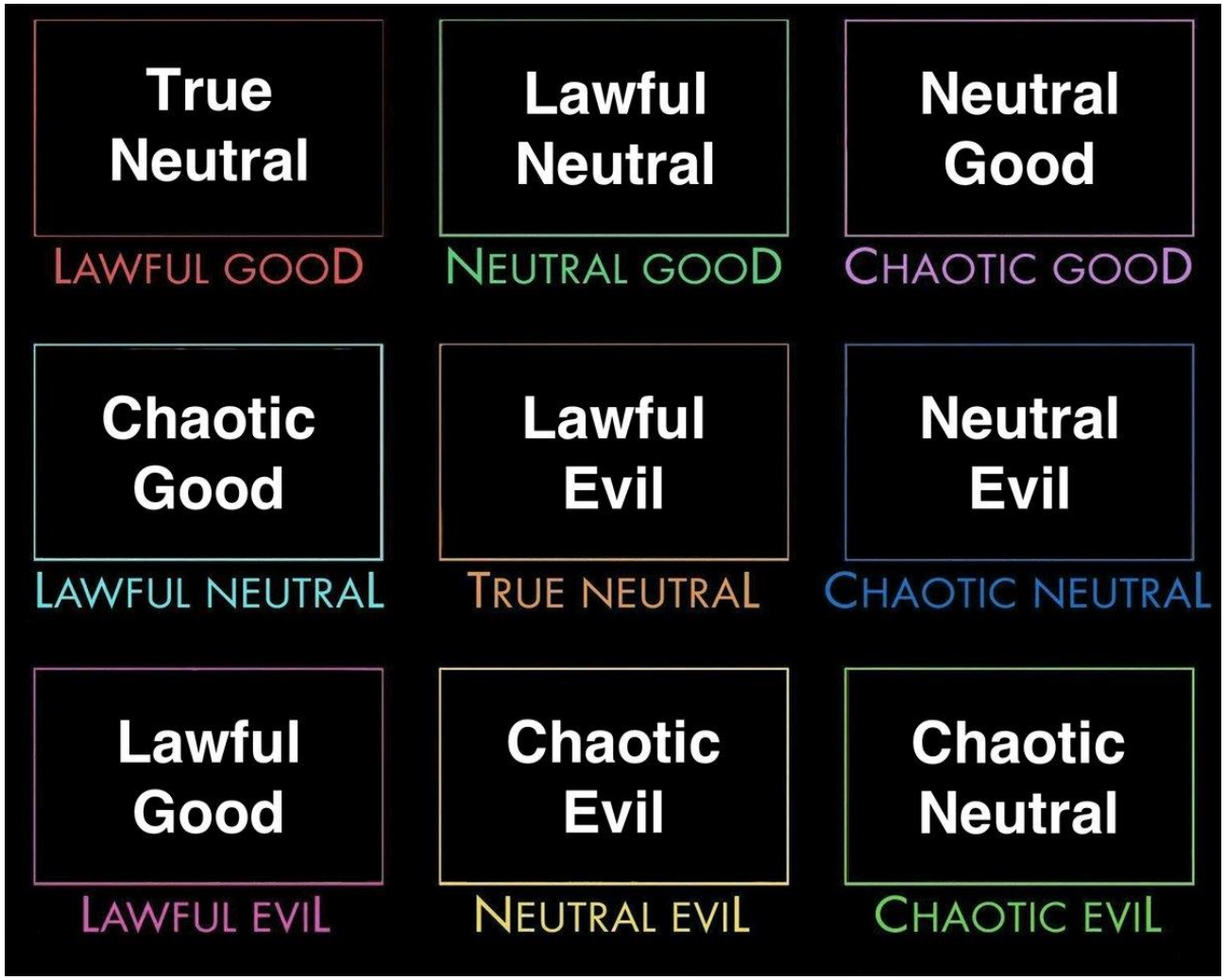
GPFS AFM “home”

TS4500 + 10 *
TS1160's
Preservation Copy 1

TS4500 + 10 *
TS1160's
Preservation Copy 2

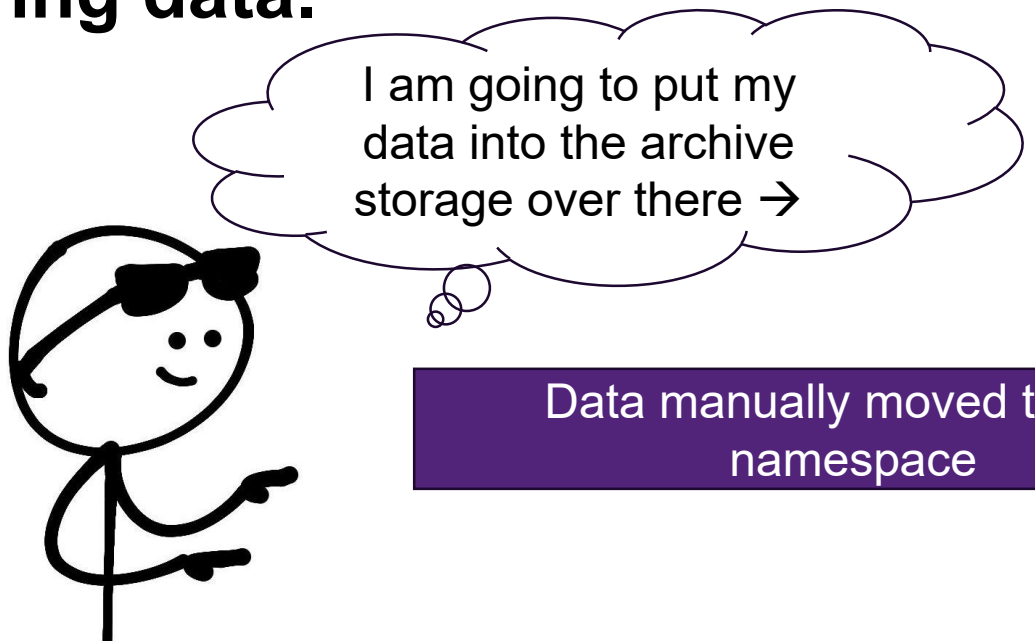
ZWS Archive Disk
Preservation Copy1
@ Polaris

The **EXPLICIT** versus **IMPLICIT** archive argument



What is an explicit archive?

- I promise it is not rude, nor impolite. Your data will still be perfectly well mannered, and this is family time-slot.
- It is when a user must make a conscious decision to put the data into a location/filesystem/storage technology that is **capable of archiving data.**

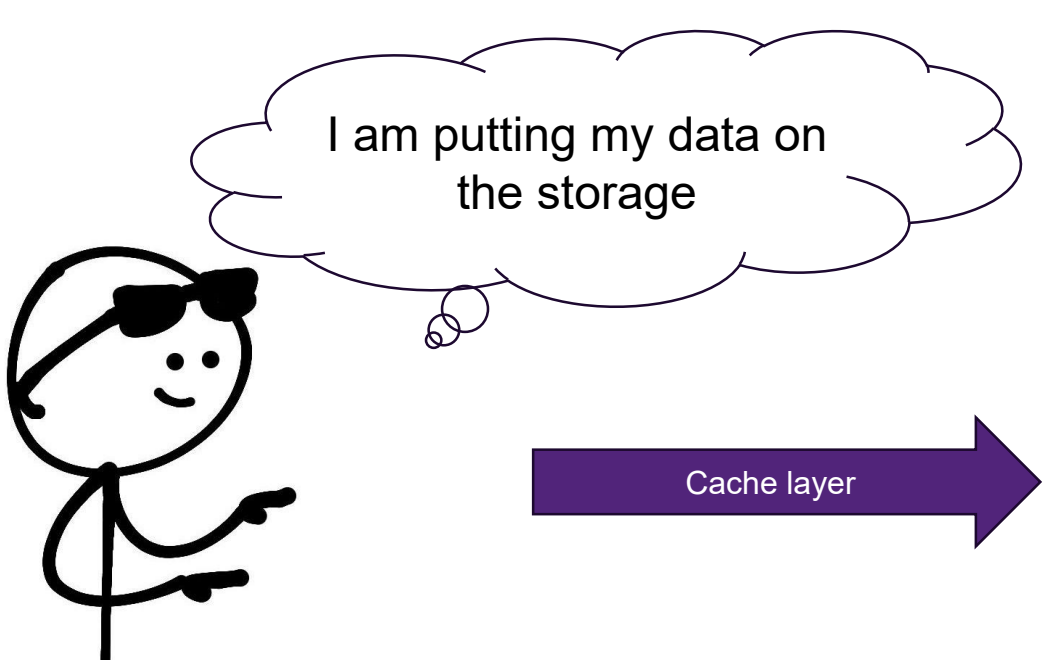


Data manually moved to new namespace

The archive filesystem

What is an implicit archive?

- It is when a user has no awareness of the attributes or features behind it. It so happens that this location/filesystem/storage technology is **capable of archiving data.** There is no deliberate choice to *"copy into that place that preserves and archives"*



Data migrates into archives, automatically

The archive filesystem

Explicit archive; +ve/-ve

The good:

- Deliberate choices of different environments mean a user responsibility shift. If they do not take responsibility, they do not archive.
- **Perceived** lack of complexity. Two systems. Nothing to do with each other.
- Less moving parts, technology-wise.

The bad:

- You're creating a data "silo".
- You're creating a new namespace.
- You're potentially breaking integrity of data.
- You're relying on users to take responsibility.
- If done badly – performance impact.

Implicit archive; +ve/-ve

The good:

- Users just use their (one) filesystem. It's a data-mesh/data lake that they just expect to see and it exists as a single point of presentation.
- Governance namespace is small. Can control it all from one point.
- Financial benefits if done correctly.
- Data movement is programmatic and declarative. Users do not do it.
- You can "hide" the performance of an archive platform if done right.

The bad:

- More moving parts in software complexity.
- Special filesystem technologies required.
- Big networks and a lot of bandwidth required to do this properly.

[BTW, there is a great Vegetarian Middle Eastern place not far from here…]

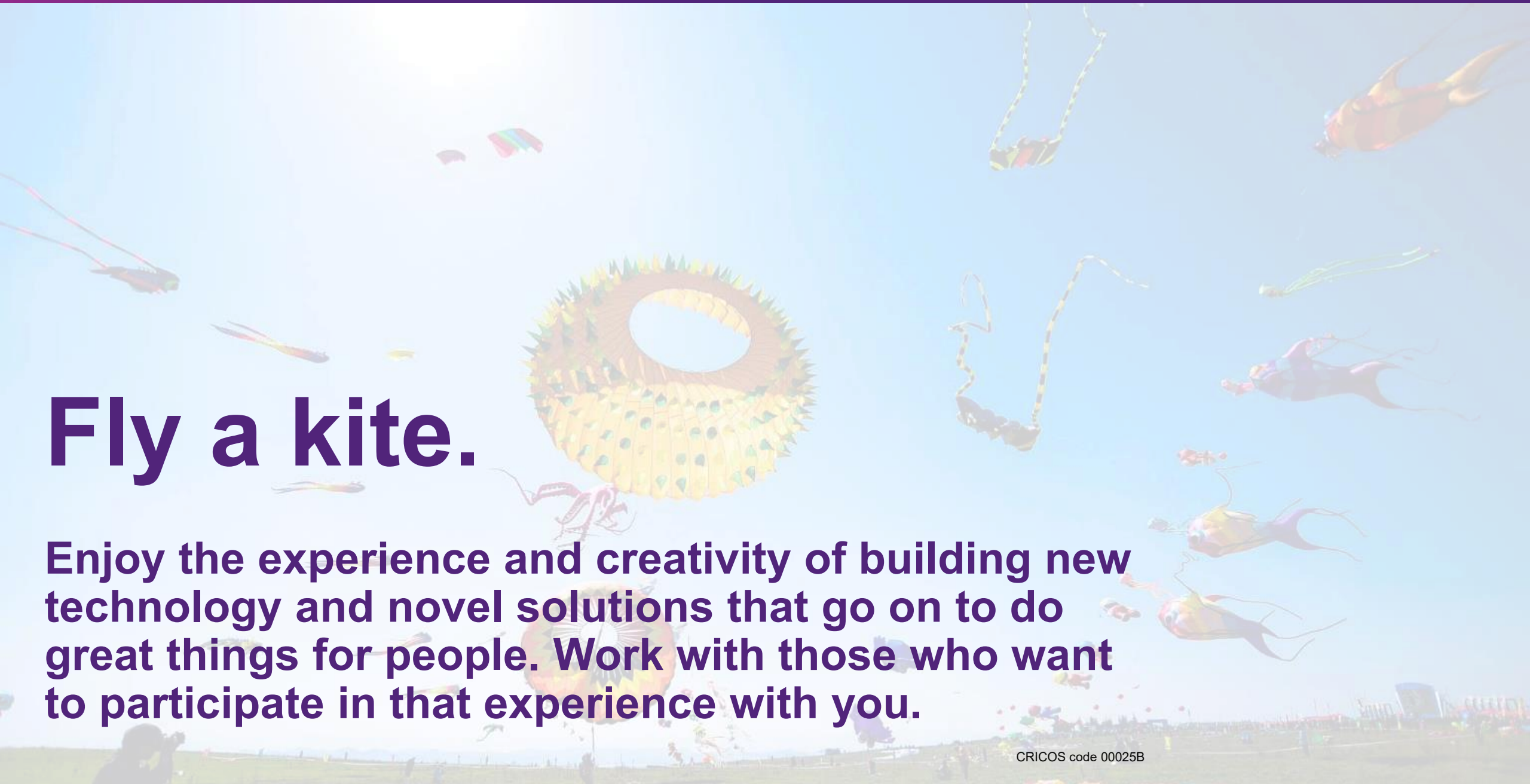
It wasn’t "just the vendor". This was the technology and development partner. Needed to work fluidly and as one.

People made this successful.

There were bad days.
We did our level best to not let our *users* have bad days.

Innovation is hard. We didn’t get into it because it was an easy win.

It was and still is a long game.



Fly a kite.

Enjoy the experience and creativity of building new technology and novel solutions that go on to do great things for people. Work with those who want to participate in that experience with you.

Thank you.

HPE:

- Kirill Malkin, Sergey Shlepakov, Jo Fryer, David Pickering, Gerald Hofer, Cihan Altinay, Helen Roberts, Susheel Gokhale, Zsolt Ferenczy, Sudha Srinivasan

ex-HPE:

- - Nick Conomo, Mark Wiertalla

IBM:

- Chris Maestas

UQ:

- David Abramson, Ashley Wright, Michael Mallon, Owen Powell, Rowan Salt, Stephen Ross

ex-UQ:

- - Rob Moffatt