Met Office

The forecast is cloudy

Jeremy Tandy | Principal Fellow, Met Office | 30-Oct-2025

Met Office: weather meets technology

National Infrastructure, Global Impact
The Met Office is the UK's national meteorological service, delivering critical weather and climate intelligence that underpins public safety, economic resilience, and international collaboration.

Data at Unprecedented Scale
We operate one of the world's most advanced environmental data pipelines—ingesting, processing, and publishing terabytes of observation and simulation data every day.

Cloud as a Strategic Enabler
Cloud technologies are transforming how we manage, optimise, and share data—enabling elastic scalability, on-demand compute, and proximity to global users and partners.

Interoperability and Open Standards
We champion open data and interoperable systems, contributing to global initiatives like World Meteorological Organization's WIS2 to ensure seamless, federated access to authoritative environmental data.

Cloud characteristics

On-demand self-service
Users can provision services, such as compute time or storage, as they require them using automated tools without the intervention of humans from the cloud provider.

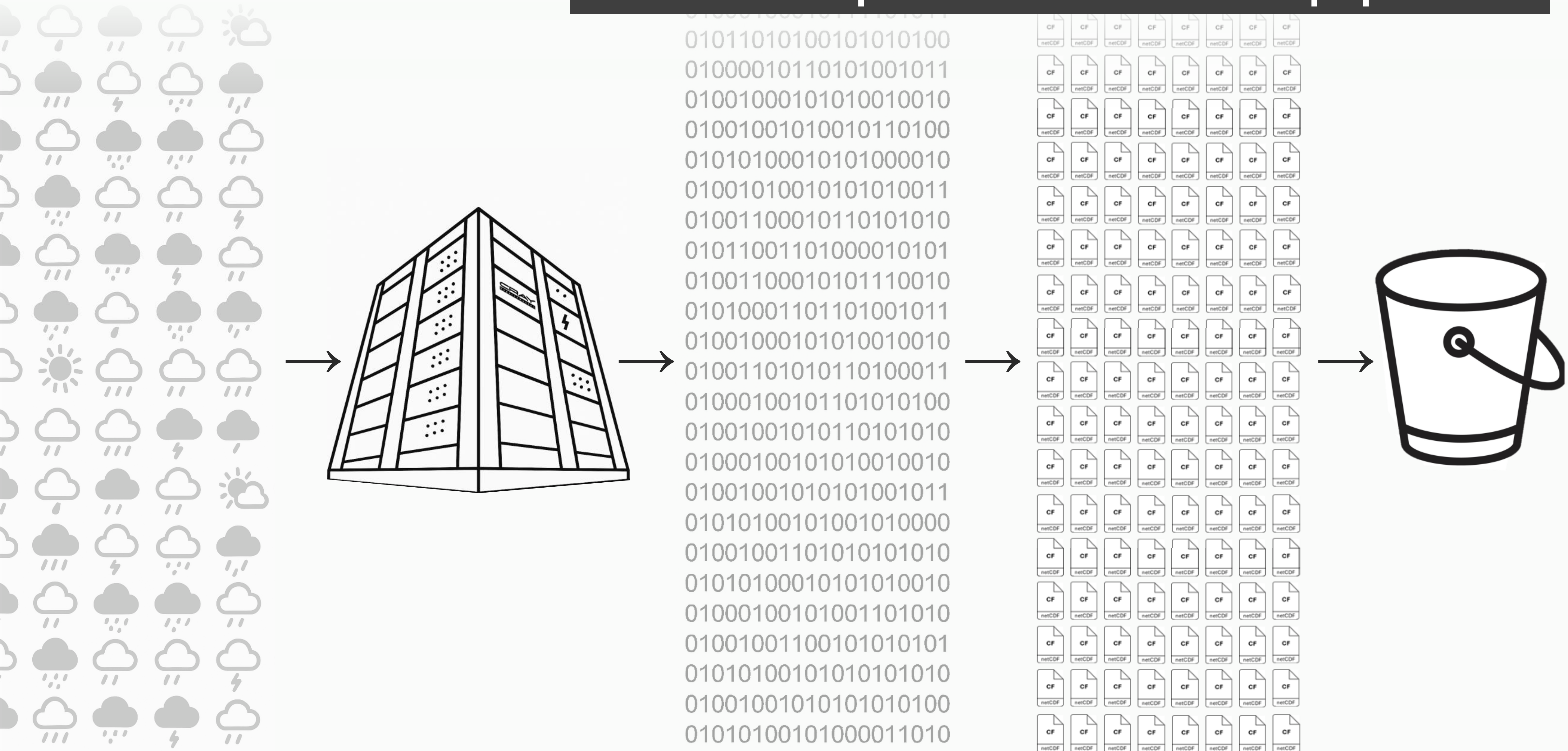
Leverage pooled resources
The services offered by the cloud provider share a pool of resources, such as processors or storage, which are opaque to the consumers. Actual location or specific identity are abstracted away from the consumer. Capacity management, pooling and time-sharing management are the responsibility of the cloud provider to assure the cloud services' availability and reliability.

Rapid capacity elasticity
The capacity of services can be expanded and decreased rapidly on-demand by the users, without the intervention of provider's people. For example, additional processing power during peak periods of Web-site usage.

Support multiple sandboxed systems
Systems operating on the cloud are securely independent and non-interfering even though they share a pooled resource base.

Unified access to metered services
There are a common set of credentials for accessing all the services available to a user (one login), and the usage of those services can be monitored in near-real-time as a usage-based metered services

Weather prediction data pipeline



Data value comes from its (re-)use

So, make the data more useful

1 Don’t move (all) the data

- cloud-optimize your data for power users*
- offer HTTP APIs for common services**
- let users know that your data exists***

*   you’re a power-user too – build your own services over the cloud-optimized data

**   visualise with OGC-API Maps
subset (Earth-system) data with OGC-API EDR
compute statistics on-the-fly (etc.) with OGC-API Processes

***   publish discovery metadata with OGC-API Records
publish real-time notifications with OGC-API PubSub (draft)

1 Don’t move (all) the data

- cloud-optimize your data for power users

2 Provide proximate compute

Met Office

Provide proximate compute

Why so close?

Applications using or processing data need to read it into memory.

Data read speed is affected by network bandwidth and geographic distance.

Processing data close to where it is stored will improve application performance.

If we don’t move (all) the data – we need to compute where the data is published.

Compute in parallel, scale elastically

Self-service, on-demand, elastic provisioning means that:

- you can meet demand from a large user-base; and
- applications can turn a time-bound problem (1 core x 1000 seconds) into a resource-bound problem (1000 cores x 1 second).

High-performance data sharing depends on a cloudy foundation.

1 Don’t move (all) the data

- cloud-optimize your data for power users

2 Provide proximate compute

3 Objects not files

Objects not files

Block store vs filesystem vs object store

Cloud providers offer several varieties of storage – each with their advantages.

- Block stores provide high-speed, low-latency data IO with raw block access.
- Filesystems support legacy applications.
- Object stores prioritize scalability and throughput.

Use cloud object stores for scalable data workflows

Scalability: object stores are designed to scale horizontally to exabytes of data.

Durability and availability: data can be replicated across multiple (geographic) zones with very high durability and automated integrity checking (e.g., S3 provides “11-nines” durability).

Cost effective: service providers may offer tiered storage (e.g., hot, cool, archive) to optimise cost based on access patterns.

Access via HTTP APIs: objects (and byte-ranges within objects) are accessed via RESTful APIs making them ideal for distributed processing, serverless workflows (e.g., Spark, Dask, ML pipelines).

Event-driven integration: support for triggers on object creation or modification – useful for automating workflows.

Met Office

1 Don’t move (all) the data

- cloud-optimize your data for power users

2 Provide proximate compute

3 Objects not files

4 Use cloud-optimized formats

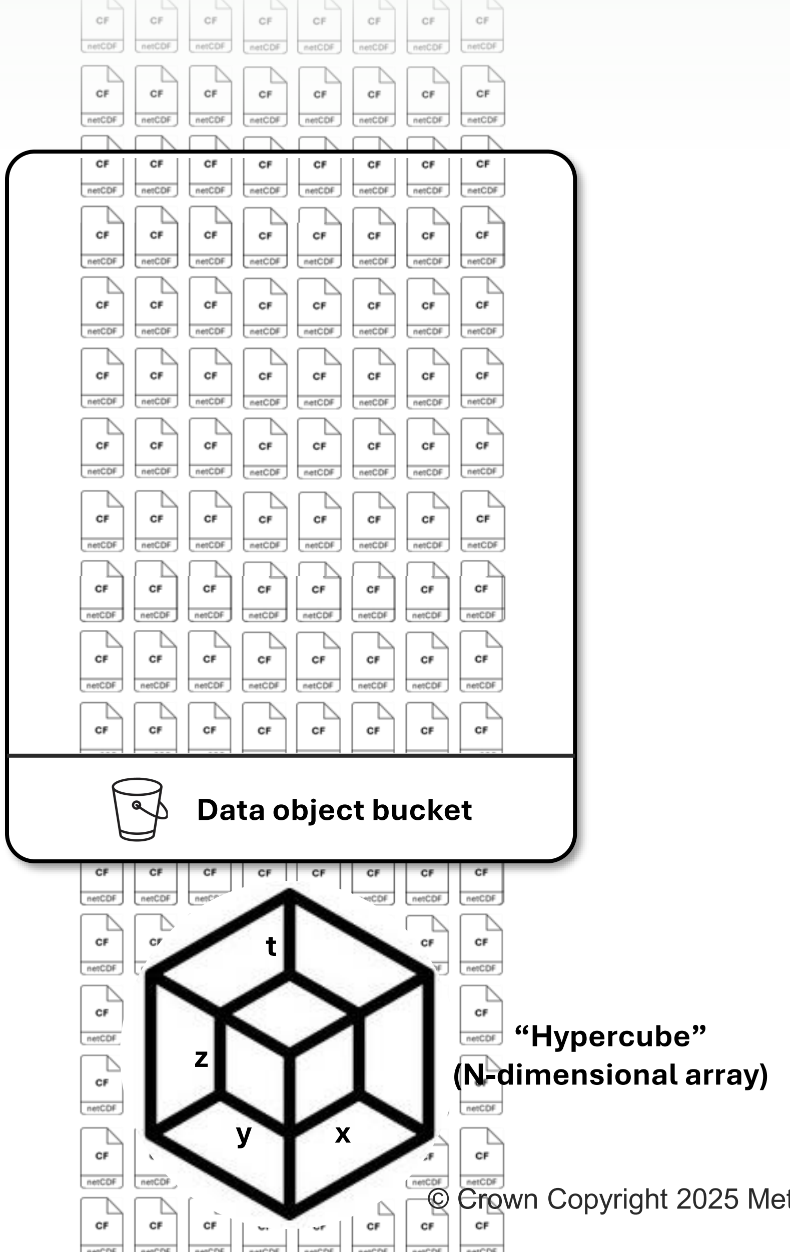
Use cloud-optimized formats

Traditional approach

- Weather prediction model output is a “hypercube” – a set of N-dimensional arrays (x, y, z, t) for a long list of physical parameters.
- Data chunked into (000s of) discrete files – breaking a large dataset into smaller, more manageable, individually compressed pieces.
- Format-specific headers contain metadata about the content.
- Local storage – fast to index metadata to determine what each file contains (low-latency file reads).

Drawbacks?

- With object storage latency is comparatively high – accessing every object to read metadata is inefficient (slow).
- Users must manage the overhead of working with 000s of objects.

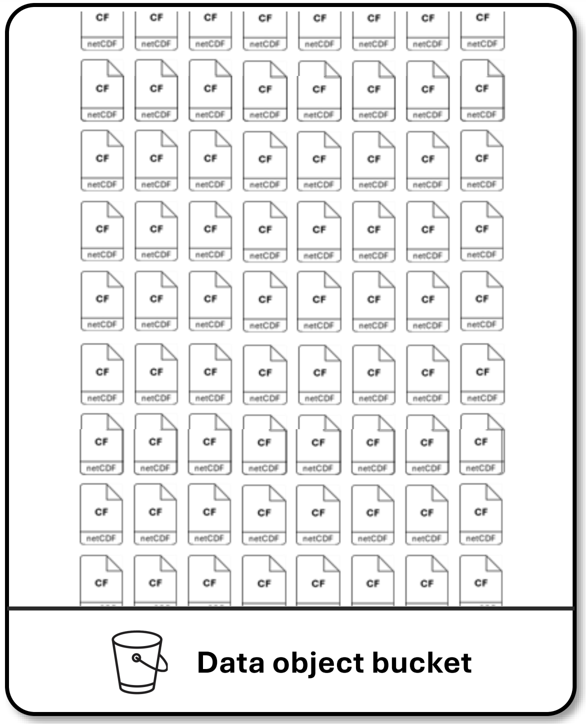


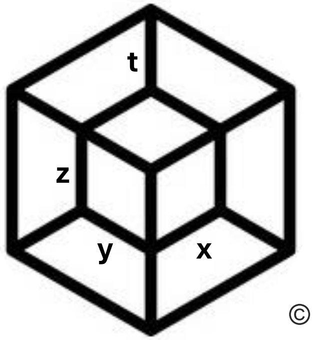
Met Office

Use cloud-optimized formats

Cloud-optimized?

- Designed for object storage with HTTP (range) requests – high throughput, (comparatively) high latency.
- Treat the whole dataset as a single entity (i.e., the hypercube as an addressable object).
- Minimize the number of reads needed to determine location of the bytes you need (i.e., metadata).
- Hide the chunking complexity from the user (objects, internal tiles, or both) – now the chunking doesn't have to be a compromise of read-performance (small ⇒ many) vs file-management (fewer ⇒ large).
- Lazy-load – use metadata to build an "empty" data structure in memory (a "virtual hypercube"), parallelize reads to get only the bytes you need when they're needed.
- Provide libraries to make application development and data analysis simpler.





"Hypercube"
(N-dimensional array)

Met Office

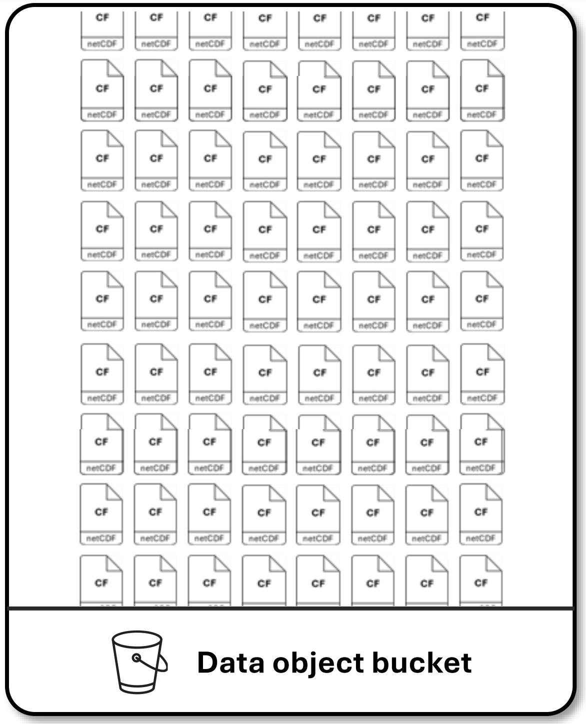
Use cloud-optimized formats

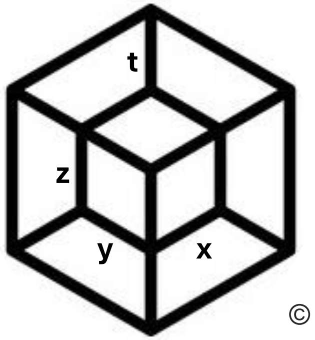
xarray
https://docs.xarray.dev/en/stable/#

Zarr
https://zarr.dev/

VirtualiZarr Public
https://github.com/zarr-developers/VirtualiZarr

NETCDF
https://www.ogc.org/standards/netcdf/





"Hypercube"
(N-dimensional array)

Choosing the right format?

- Zarr is used for storage of large N-dimensional arrays (tensors) enabling efficient IO for parallel computing applications.
- GeoZarr (draft) aims to provide geospatial extensions to Zarr – conventions for storing multidimensional georeferenced data.
- Xarray (python) reads and writes Zarr directly and supports lazy-loading.
- Zarr does have a native binary format – but VirtualiZarr (and Kerchunk) creates "virtual" Zarr datacubes by indexing legacy formats like netCDF, HDF5 and GRIB, even when compressed.
- The "virtual" Zarr provides an abstraction layer making 000s of netCDF files behave like a cloud-optimized dataset.

Use cloud-optimized formats

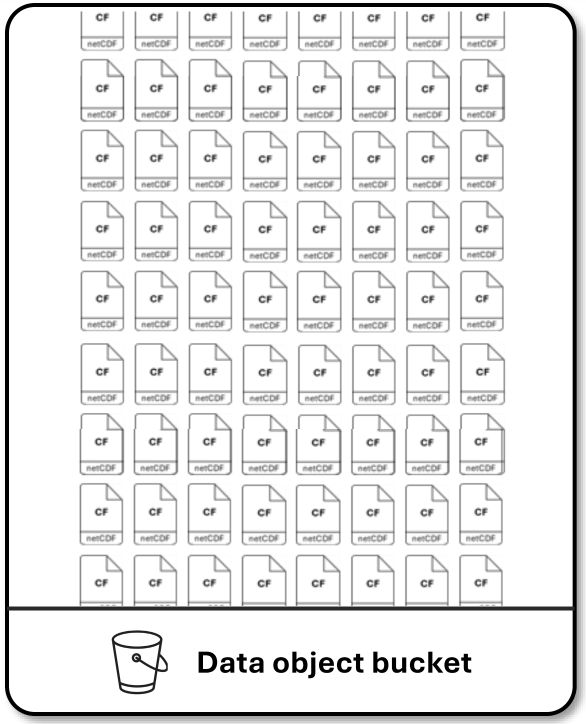
https://docs.xarray.dev/en/stable/#

https://zarr.dev/

VirtualiZarr Public

https://github.com/zarr-developers/VirtualiZarr

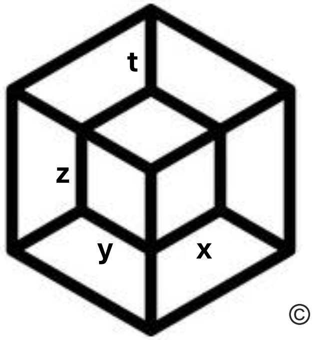
https://www.ogc.org/standards/netcdf/



CNG

https://cloudnativegeo.org/

- Points | Lines | Polygons | Rasters | Datacubes | Point-clouds
- For more information see the Cloud-Optimized Geospatial Formats Overview from the Cloud-Native Geospatial Forum (CNG).



"Hypercube"
(N-dimensional array)

Indexing cloud-optimized archives

STAC

http://stacspec.org

Zarr

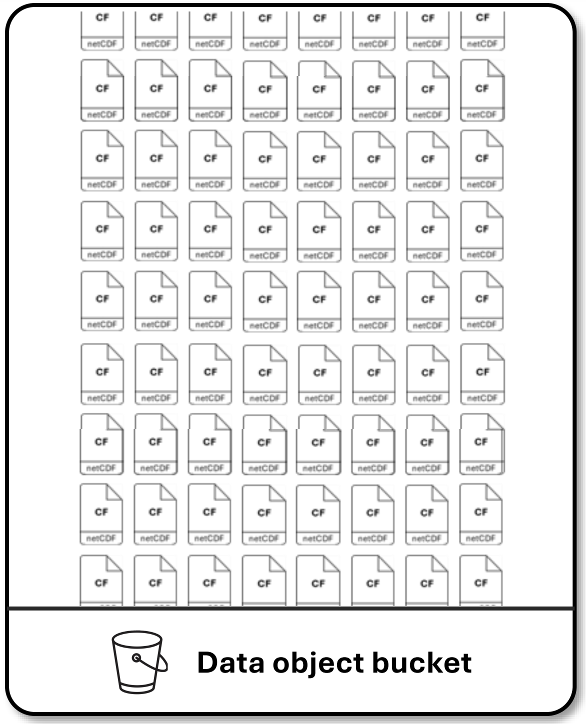
https://zarr.dev/

VirtualiZarr Public

https://github.com/zarr-developers/VirtualiZarr

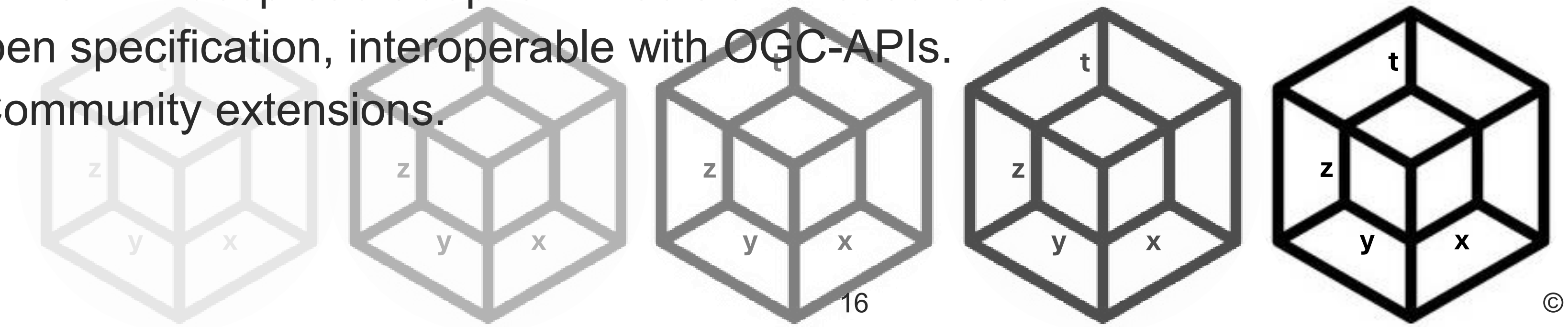
NETCDF

https://www.ogc.org/standards/netcdf/



Spatio-Temporal Asset Catalogue

- Aim: Provide a simple mechanism for providers of spatio-temporal assets (Imagery, SAR, Point Clouds, Data Cubes, Full Motion Video, etc) to expose their data in a common way so that new code doesn't need to be written whenever a new data set or API is released.
- Simple: Hypermedia + GeoJSON + (optional) API.
- Community-driven: Widespread adoption – tools and resources.
- Standard: Open specification, interoperable with OGC-APIs.
- Extensible: Community extensions.



“Hypercube”
(N-dimensional array)

Cloudy data access example

Simple serverless (meta)data driven data viewer

no “moving parts” + cloud-optimized data = low-cost scalability

Lambda
matplotlib &
other libraries

OGC-API Maps

Render the data and serve with some UI components

Approx cost per request: ¢0.02

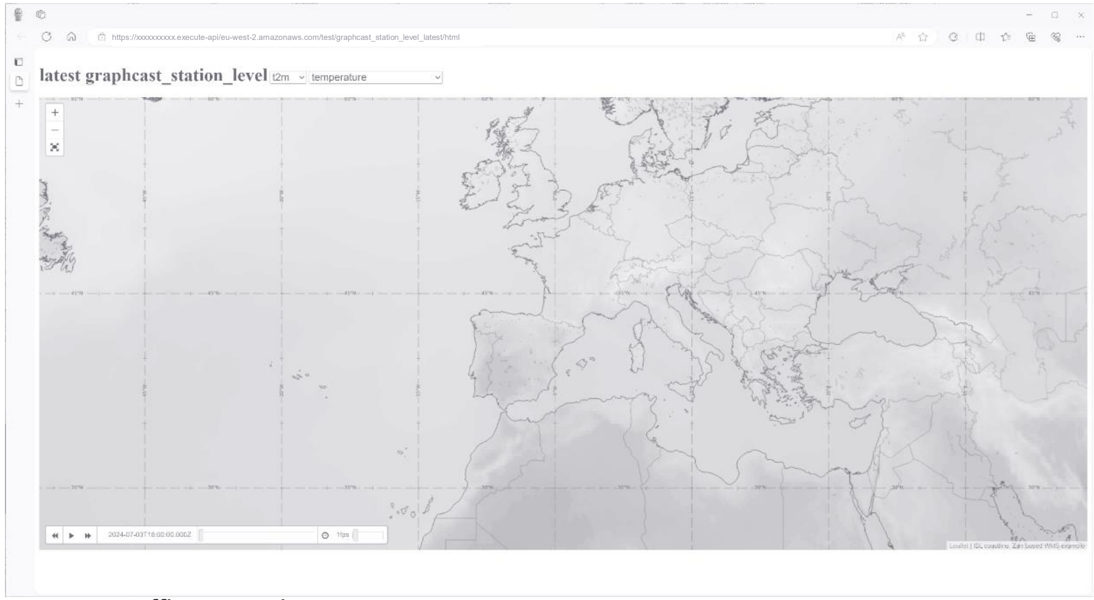
Grab the data “assets” from the S3 bucket (parts thereof)

S3: multi-petabyte data archives

Read the STAC metadata to find the data objects needed for the request

STAC: Human-browseable and machine-readable metadata (static JSON resources)

http request



Cloud provisioning models

Private cloud
For the exclusive use of a single organisation. They are managed and operated by the organisation itself, or a third-party – and may exist on or off premises. Most frequently this model is used as a new way to provide IT services within an organisation by sharing existing hardware/software infrastructure, breaking the need to make capital investments on a per-system basis.

Community cloud
For the exclusive use of a community of consumers with shared concerns or need to collaborate. They are managed and operated by one, or several, of the consumer organisations, or a third-party on the community's behalf. They may exist on or off premises.

Public cloud
For openly accessible use by the public. They are managed and operated by businesses or governmental organizations. They exist on the premises of the cloud provider.

Sovereign cloud
Ensures that data stays within a specific country or legal jurisdiction, is governed by local laws, and is operated by trusted entities — giving organisations full control over data residency, access, and compliance.

Hyper-scalers
Are commercial operations such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud, whose public clouds are of such a scale that they effectively offer consumers unconstrained capacity.

Met Office

Cloud provisioning models

Private cloud
For the exclusive use of a single organisation. They are managed and operated by the organisation itself, or a third-party – and may exist on or off premises. Most frequently this model is used as a new way to provide IT services within an organisation by sharing existing hardware/software infrastructure, breaking the need to make capital inve

But there’s no such thing as a free lunch

Communi
For the exclusive use of a community of consumers with shared concerns or need to collaborate. They are managed and operated ... on the community’s behalf. They may exist ...

Someone always pays €€€€

Public cloud
For openly accessible use by the pu ... businesses or governmental organizations. They exist on the pre

CAPEX vs OPEX

Sovereign cloud

FinOps: balancing cloud costs and business value

Hyper-scalers
Are commercial operations such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud, whose public clouds are of such a scale that they effectively offer consumers unconstrained capacity.

Scaling with open data initiatives

Open data ⇒ Unconstrained demand ⇒ €€€€

Could you share the load with Big-Tech CSR or philanthropists?
(but don't forget those cloud-optimized formats ☺)

The Amazon Sustainability Data Initiative
ASDI

https://sustainabilityexchange.amazon.com/data-initiative

https://www.metoffice.gov.uk/services/data/external-data-channels#ASDI

Microsoft
Planetary Computer

https://planetarycomputer.microsoft.com/

Google Earth Engine

https://earthengine.google.com/

SOURCE COOPERATIVE

https://source.coop/ | https://docs.source.coop/

Questions for sustainable cloud ops

Who are your users?

Which cloud platform(s) do they prefer?

What is their primary access pattern?

How will you measure business value?

Do you need your own data platform?