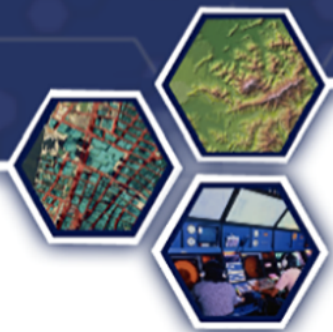**Sponsors**

WFS 3.0 Hackathon
6 – 7 March 2018

Sponsors

Static WFS

aka Simple Features for the Web
Some thoughts by Chris Holmes

WFS 3.0 Hackathon
6 – 7 March 2018

Background

- In October in Boulder we defined the SpatioTemporal Asset Catalog (STAC)
- At the core it is very similar to WFS, but each feature refers to an Image (or more generically an 'Asset')
- One powerful idea that emerged was a 'Static Catalog', that makes data accessible, though not queryable
- Is simply a crawlable link structure in JSON
- It lowers the barrier to expose one's data, and makes for an incredibly reliable catalog
- Other more active API's can crawl it and serve up the data for more advanced queries

OGC®

Alignment with WFS

- A major goal for this week is to align STAC with WFS, so that a SpatioTemporal Asset Catalog is simply a slightly more constrained profile of WFS
  - Add in required content elements (time, provider, license), make some simplifying choices (no XML, pick a query language, etc)
- But there is as of yet no well-defined WFS equivalent to a Static STAC.
- This might be outside of the scope of 'WFS', as it perhaps takes too much away. But ideally it's at least a complementary specification that is 'crawl compatible'.

OGC®

Initial Static WFS Directions

- Figure out if we can make an OpenAPI document that is constrained to not allow any queries

- Define a GeoJSON-based structure with links that can be crawled (likely mirroring the 'links' section of STAC, plus some equivalent to the 'catalog' concept that is used for traversal)

- Prototype an HTML version that is a user friendly way to explore the data

- Create command line tools to create a static WFS / simple features for the web (I need help with this, my programming skills are rusty). OGR plugin eventually?

OGC®

SpatioTemporal Feature

- In defining STAC we added in Assets and 'Time'
- Ideally there is a more generic shared definition of 'Time', that STAC Items can extend
- Should be useful for both WFS 3.0 and for any Static WFS definition
- Ideally is just a 'pattern' that can be applied to most any data. If the same field definitions are used then clients will know they can use time in expected ways.
- There is likely overlap with 'updated' and 'published' fields that communicate about the state of data. This should play well with cache control headers, etc.

OGC®