



# **WFS 3.0 HACKATHON**

## **WHY AM I HERE?**

Chuck Heazel

# My Goals and Objectives

---



- Validate a compliance test strategy for WFS 3.0 implementations
- OpenAPI – does it do everything we need?
- Is WFS 3.0 correct and sufficient for our needs?
- Gather material for the WFS 3.0 Users Guide (don't throw the knowledge away)

# Compliance Test Strategy



1. You cannot test compliance with a microservice based standard
2. You can test compliance with each individual microservice
3. How to you know what microservices to test?
  - a) Access the OpenAPI documents
  - b) Validate the OpenAPI documents against the standard
  - c) For each microservice described:
    - i. Validate the microservice against the description
    - ii. Validate the microservice against the standard
4. How do you report compliance?
  - a) Report compliance for each microservice
  - b) Define “collections” of microservices which are commonly implemented together
  - c) Report compliance with a “collection”

# OpenAPI



- It is likely that OpenAPI does not do everything we need
- Two opportunities:
  - OAS 3.1 is under development
  - Define extensions to OAS
- For your convenience:
  - OAS JSON schema in YAML (Pull Request 1270)
  - OAS JSON schema in JSON (Pull Request 1236)
  - OAS UML model
- Known issues:
  - Inadequate support for security (issues 1473, 1464, 1437, 1416, 1393, 1366, 1229, 1004, 971, 846, 785, 680, 642, 621, 620, 585 (meta issue), 551, 550)
  - HTTP compression (issue 738)
  - Web Sockets (issue 523, 550)

# Sufficiency



- Target deployment = a WFS 3.0 front-end to a WFS 2.0 Basic implementation
- Question: Can this be built, will it work?
- Required:
  1. Transformation from OWS Service Metadata to OpenAPI
  2. WFS 3.0 Core
  3. WFS 3.0 Feature Encoding extension
  4. WFS 3.0 GML Extension
- For your convenience:
  - Initial mapping of OWS Service Metadata to OpenAPI
  - Initial mapping of WFS 2.0 to WFS 3.0

# WFS 3.0 Users Guide



- The WFS 3.0 standard will be concise and short
- This should make it more attractive to non-geospatial developers
- However, a lot of knowledge is assumed in that approach (what is a feature?)
- The WFS 3.0 Users Guide is a non-normative source for that knowledge

[https://github.com/opengeospatial/WFS\\_FES/blob/master/guide/index.adoc](https://github.com/opengeospatial/WFS_FES/blob/master/guide/index.adoc)