Introduction OGC API standards

Winter School 2022

Ms. Prajwalita J. Chavan
IIT, Bombay

Overview

- About Web API
- About OGC
- Architecture
- Timeline
- OGC Web Services
- Web Services vs API
- OGC Web API
- OGC API Standards Development
- OGC API standards
- How to approach OGC API?
- Developers
- OGC members
- Domains
- Hands-on

What is Web API?

- A Web API is a powerful tool for **sharing** information and **analysis** resources on web

- It also venerable to **attack** those resources.

- APIs deliver a **representation of a resource.**

- APIs can deliver **multiple representations** (formats) of the same resource.

- API designers must take care that the **access controls** on their resources are implemented consistently across all representations.

- Web APIs are software interfaces that use an **architectural** style that is founded on the technologies of the Web.

What is OGC?

- Open Geospatial Consortium (OGC) – **Not-for-profit**, **international** voluntary consensus standards organization

- **Implementation Standards** are different from the **Abstract Specification**.

- They are written for a more **technical audience** and detail the **interface structure** between **software components**.

- An interface specification is considered to be at the **implementation level** of detail if the resulting components **plug and play** with each other at that interface.

- OGC continues to use special **events, hackathons, and sprints** to validate the OGC API draft candidate standards as those standards have already withstood **testing and suitability evaluation.**

OGC FAIR Mission

The OGC Vision:

Building the future of location with community and technology for the good of society.

The OGC Mission:

Make location information Findable, Accessible, Interoperable, and Reusable (FAIR).

The OGC Approach:

A proven collaborative and agile process combining consensus-based open standards, innovation projects, and partnership building.

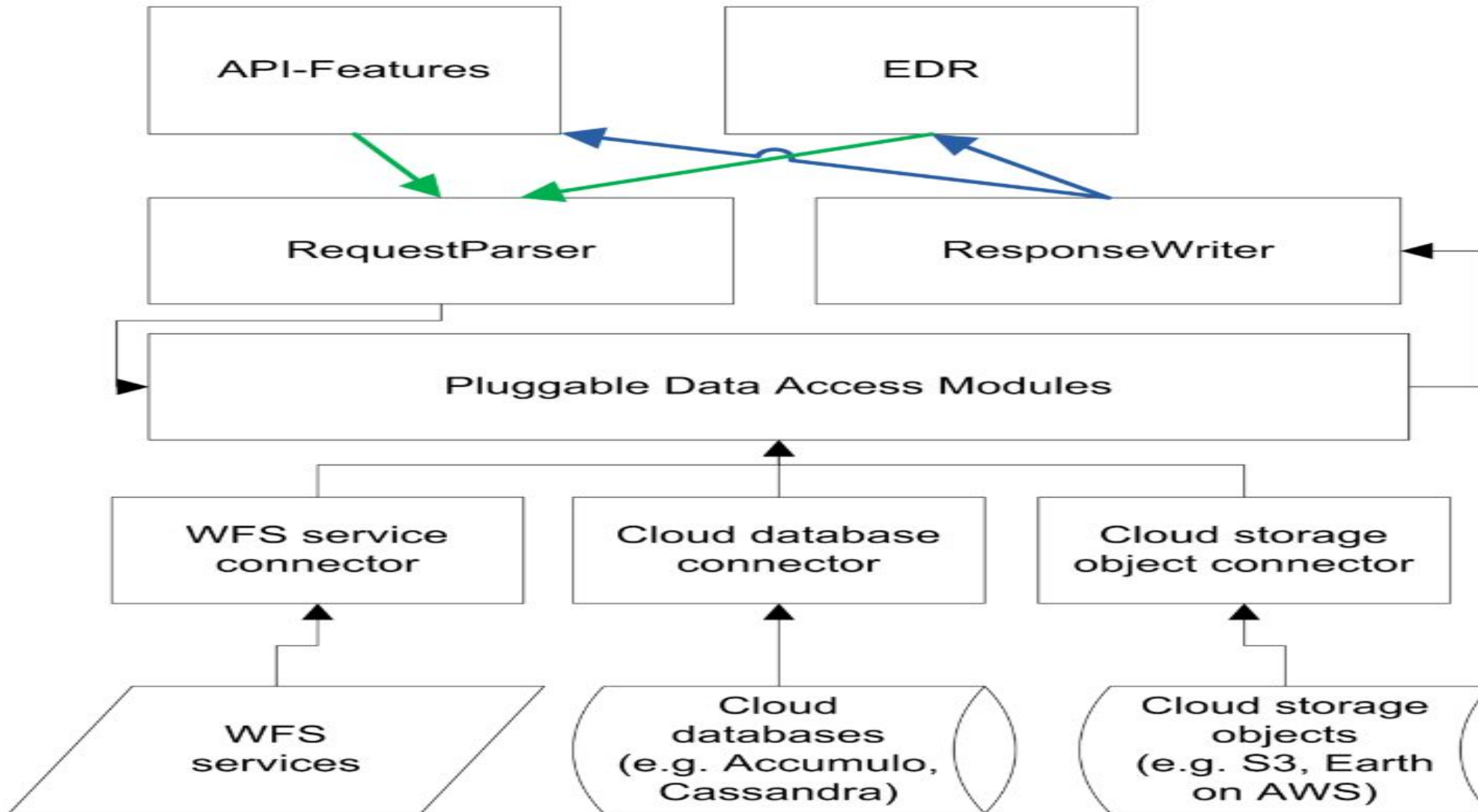**The OGC Values:**

We are open, diverse, inclusive, and accessible.

We value our technical excellence and innovation.

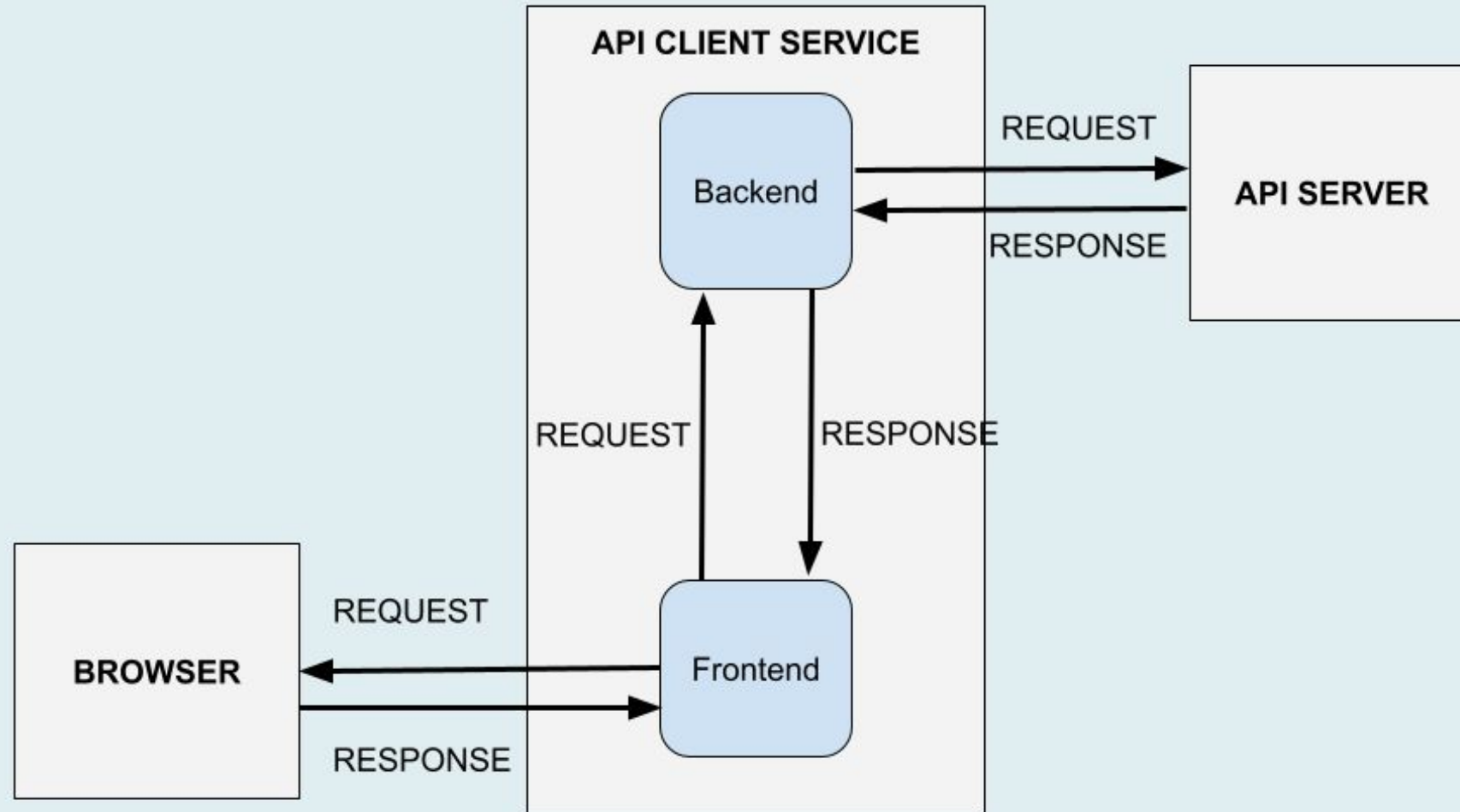*We are passionate about the greater good.*

We are community-driven and we are fair, respectful, and responsible to our members.

We always honor our commitments and aim to exceed expectations.

Architecture

Client Server Architecture

Timeline

2015
- Testbed-11 Comparison of REST to classic OGC Web Services

2016
- Testbed-12 work on a RESTful binding of the WPS
- Focus of discussions shifts from REST to Web APIs

2017
- OGC® Open Geospatial APIs - White Paper published

2018
- Work on version 3 of the Web Feature Service (WFS3) starts

2019
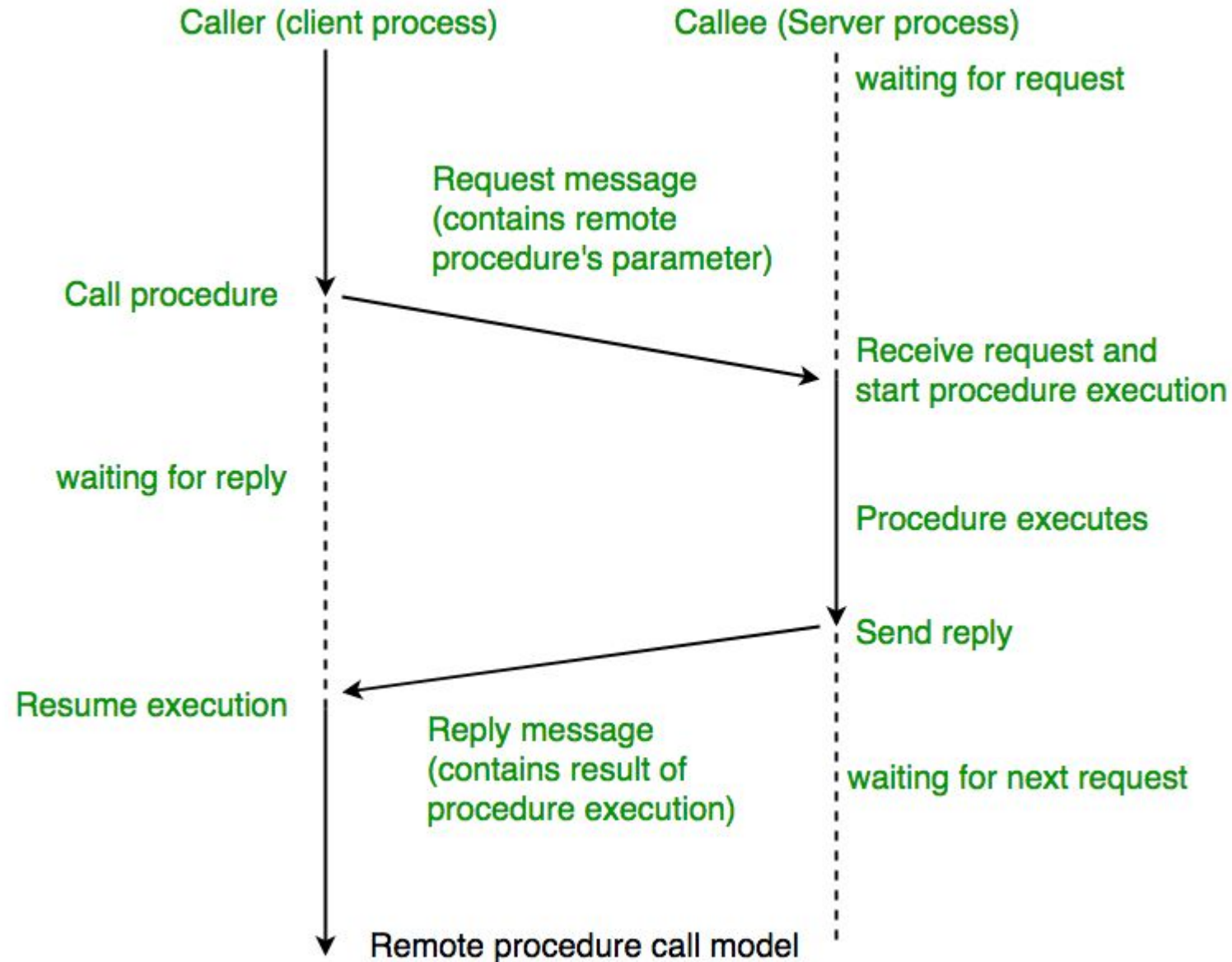- WFS3 draft specification renamed OGC API - Features
- OGC API – Features – Part 1: Core standard is published

2022
- Testbed-17: Implementing OGC API

OGC Web Services

OWS standards have historically implemented a
Remote-Procedure-Call-over-HTTP architectural style using Extensible
Markup Language (**XML**) for payloads.

- Web Feature Service (WFS)

- Web Map Service (WMS)

- Web Map Tile Service (WMTS)

- Web Processes Service (WPS)

- Web Coverage Service (WCS)

Remote-Procedure-Call-over-HTTP architecture

Caller (client process)

Callee (Server process)

waiting for request

Request message
(contains remote
procedure's parameter)

Call procedure

Receive request and
start procedure execution

waiting for reply

Procedure executes

Send reply

Resume execution

Reply message
(contains result of
procedure execution)

waiting for next request

Remote procedure call model

OGC Web Services

	WMS	WFS	WCS	WPS	SOS	SPS	CSW	WMTS
Use HTTP methods explicitly.	Y	N	Y*	N	N	N	N	Y
Be stateless.	Y	Y	Y	Y	Y	Y	Y	Y
Expose directory structure-like URIs.	N	N	N	N	N	N	N	Y
Use HTTP Error codes	N	N	N	N	N	N	N	N
Transfer XML, JavaScript Object Notation (JSON), or image.	Image	XML	Any	Any	XML	XML	XML	Image

SOS: Sensor Observation Service

SPS: Sensor Planning Service

CSW: Catelogue Service

WFS: Web Features Service

WCS: Web Coverage Service

OGC Web Services

The OGC **WMS and WMTS**: standards for map and the capability to create and distribute maps at a limited resolution and size.

- **WMS:** the **number of rows and columns** that a map should have can be **selected by the user** within limits.
- **WMTS**: the number of rows and columns of the tile is **predefined** in the tile matrix.

Web Services vs API

- OGC APIs are intended to be **simpler and more modern**

- OGC API is a **RESTful approach**, implementations of an OGC API are **not backwards compatible with OWS** implementations per se.

- However, a **design goal** is to define OGC APIs in a way that an OGC API interface can be mapped to or used as a **façade to an existing OWS** implementation (where appropriate).

- It is possible to implement a Web API that concurrently **conforms to conformance classes** from the Features, Coverages, Maps, Tiles, and other future OGC API Standards.

Web Services vs API

- A RESTful Web API style is **resource-oriented instead of service-oriented**

- Resources are over **data models**, services are over **functional modules**

- Resources is used to provide **CRUD operations**, services is used to **link modules at run time**

- Resources **insulates API consumers from changes to data models**, services allows **drop in replacements of modules, simplifying deployment and customization**.

Web Services vs API

- An application built with a **Service Oriented Architecture** is more a **'Facade'**, It combines or composes its outgoing functionality based on **functionality that is in the services** it uses 'behind the screens' (possibly over the network). E.g. its core processing consists of **calling external services, supplying them with parameters, and combining the results** with possibly some extra processing or algorithms for the user.

- An application built with a **Resource Oriented Architecture** does more of its processing internally (e.g. as opposed to calling external components) but uses external resources as input. E.g. its core processing consists of **retrieving static resources** and then doing more calculating **internally**.

Web Services vs API

- **Resource oriented**
  - **Communication** with **HTTP resource**
  - Involves operation for **lifecycle of resource** through **HTTP GET, PUT, POST** etc.
  - resource **data** can be **cached**

- **Service oriented**
  - Involves communication with **specific application service**
  - **doesn't involve lifecycle operations** on service lifecycle management
  - All messages are **sent to service endpoint**
  - **endpoint decides** how to process request

OGC Web API
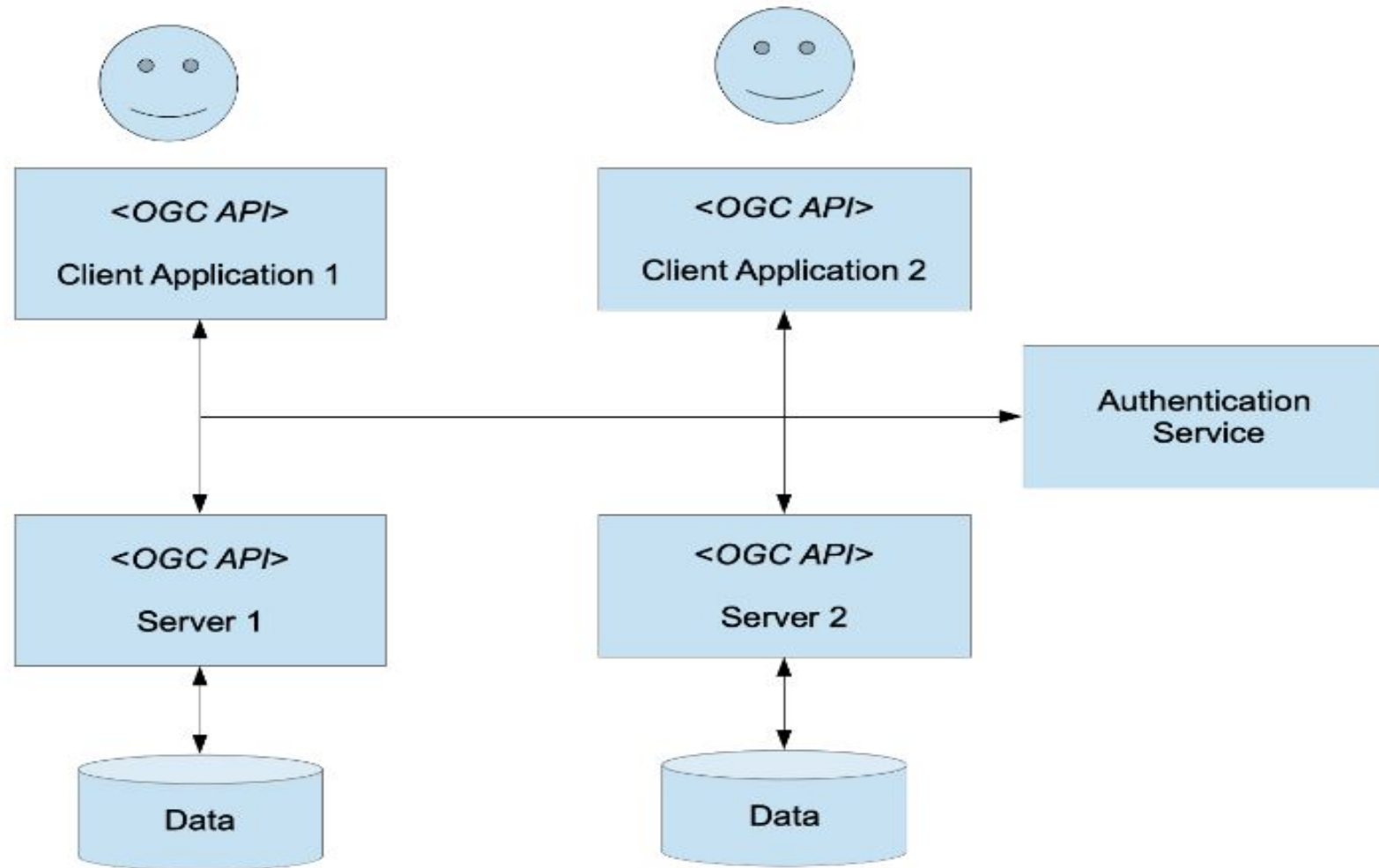
OGC Web API
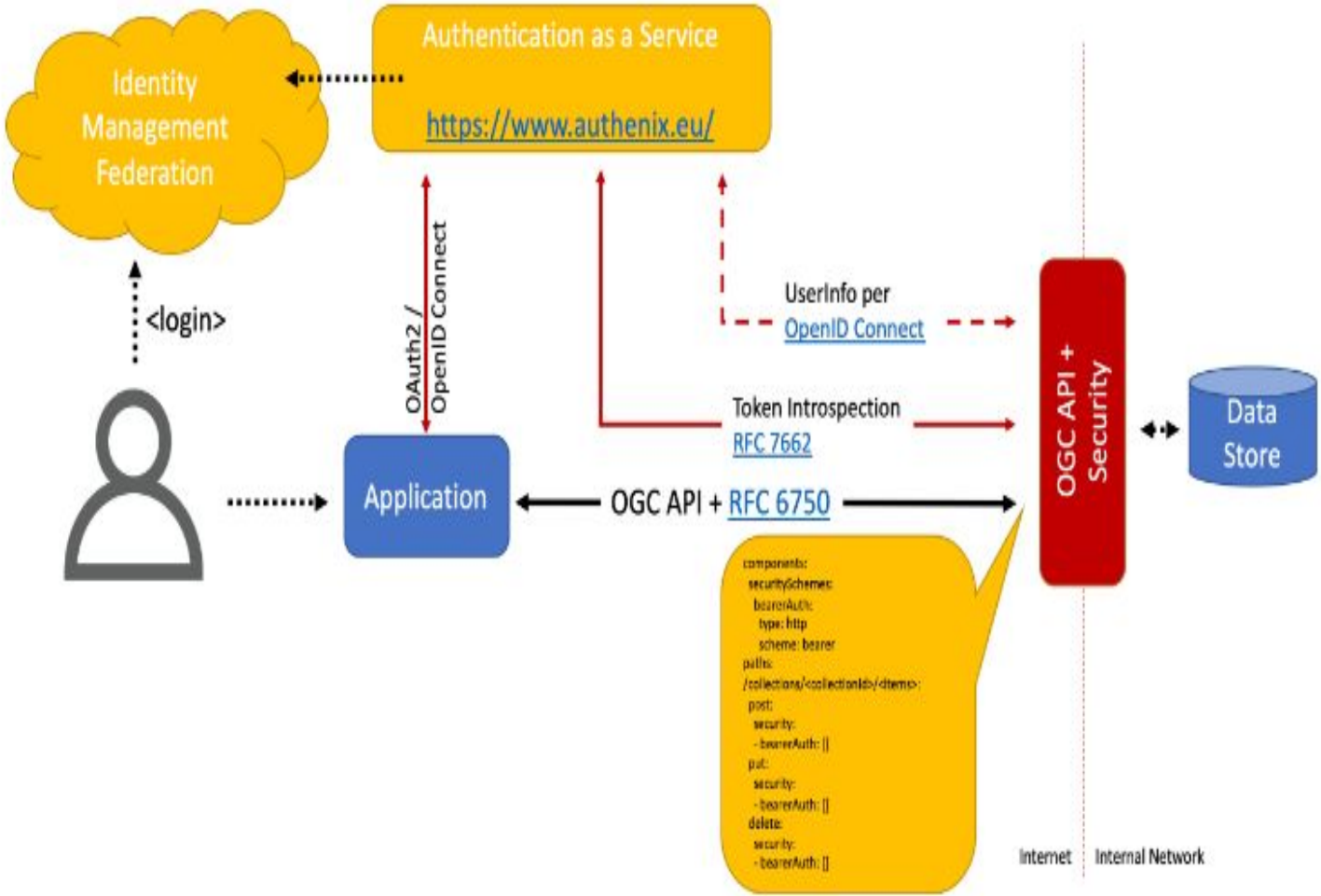
OGC Web API

- **Web API:** An **Application Programming Interface (API)** using an architectural style that is founded on the technologies of the Web

- Family of OGC standards developed to make it **easy** for anyone to **provide geospatial data to the web**

- OGC API standards: **define modular API building blocks** to spatially enable Web APIs in a consistent way

- The OpenAPI specification: is used to define the **API building blocks**.

- These standards build upon the legacy of the OGC Web Service standards (WMS, WFS, WCS, WPS, etc.), but define **resource-centric APIs** that take advantage of modern web development practices.

Open API
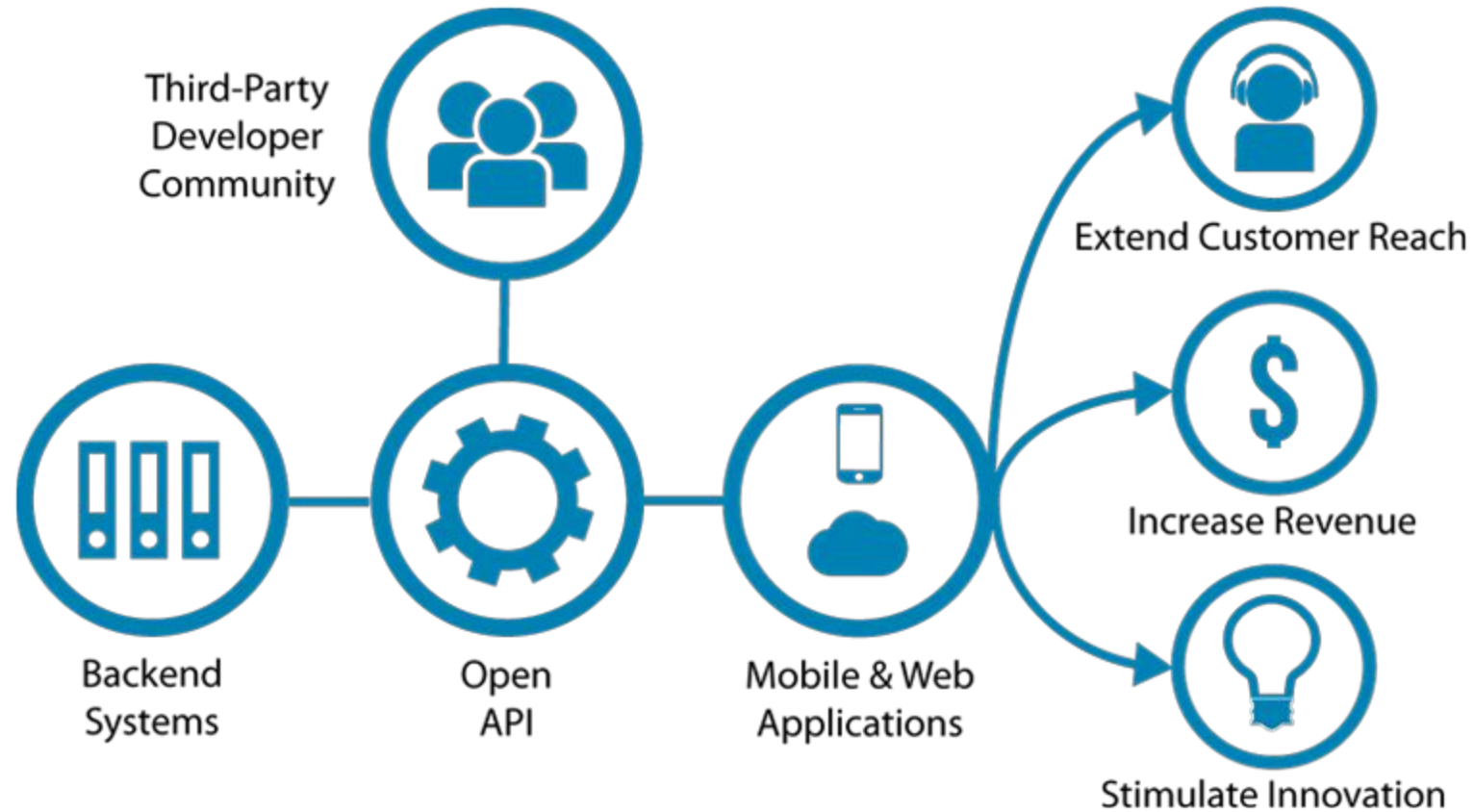
OGC Web API
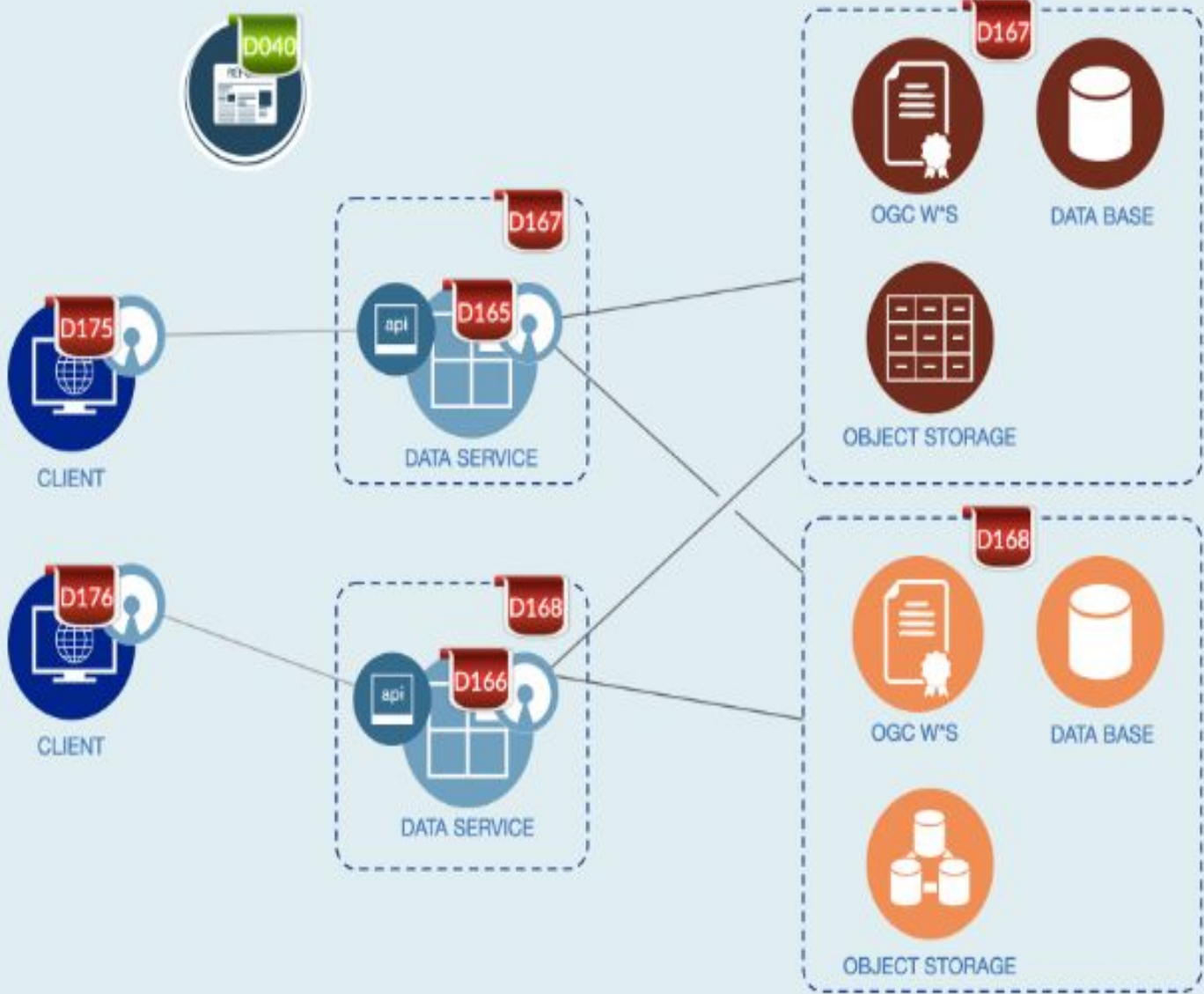
OGC Web API
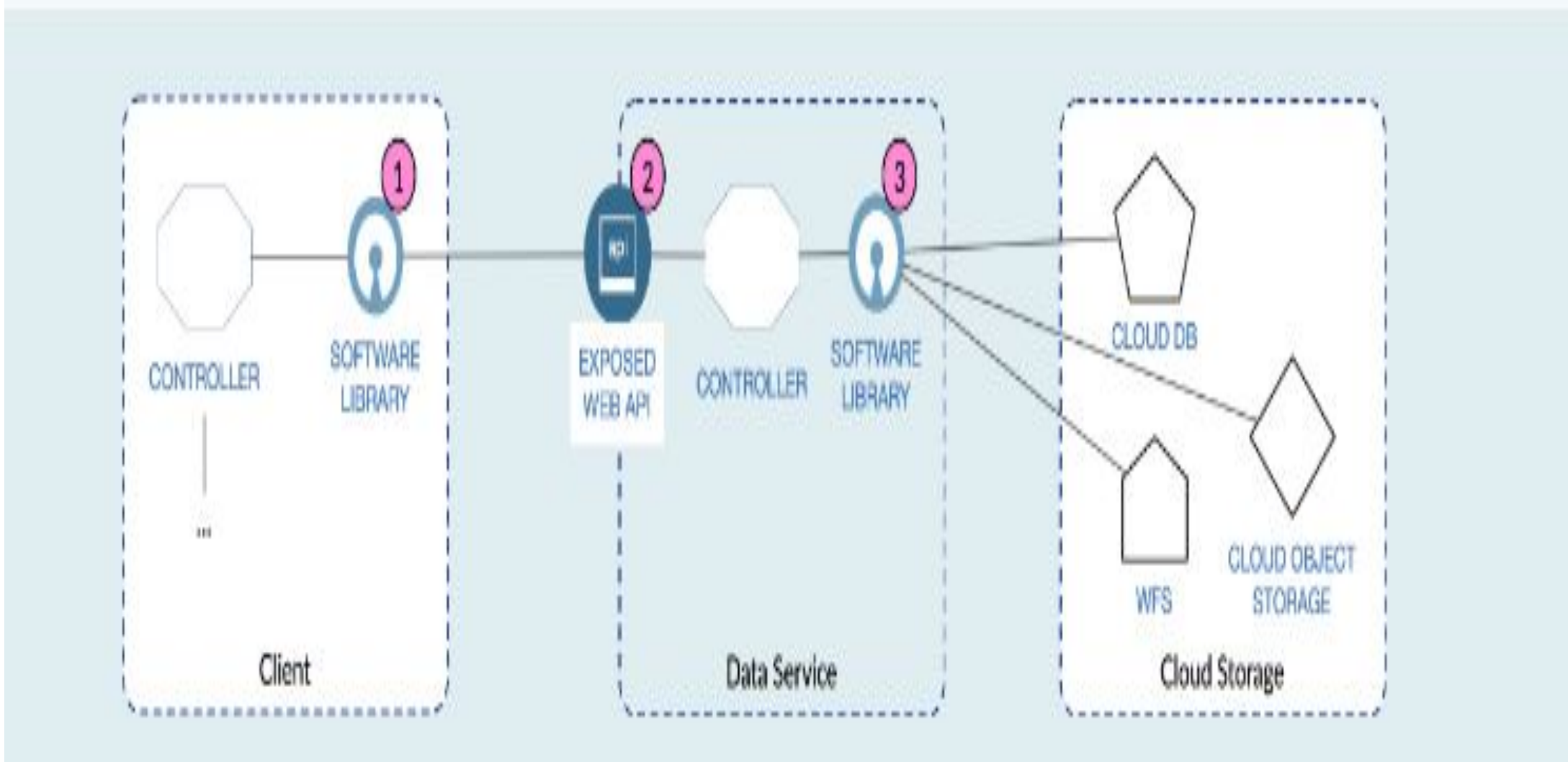
Why OGC Web API

- Advance development of **international standard**

- Inhance **information integration**

- APIs are a popular, effective method for **rapid software development**

- Open Standards enable **interoperability of independent implementations**

- OGC APIs will improve **interoperability between Web APIs**

- Reduce fragmentation and inefficiency in **data sharing**

Why OGC Web API

- OpenGIS Specifications have been agreed to by a broad swath of the entire community and are **supported by most of the geoprocessing software vendors**

- OpenGIS **links geographic data with mainstream IT**, via the geospatial architecture in which geospatial components from multiple sources can plug-and-play through standard interfaces.

- OpenGIS maintains a leading technical architecture made up of the suite of **interfaces for the benefit of the industry and its customers** and works to minimize greed, parochialism, and lethargy in the market

- Vendor implementation in products enables you, the customer, to directly access and **use data produced by programs from many vendors** -- not just one

OGC API Standards Development

- **Modular** API building blocks; spatially enable Web APIs in consistent way

- **Spatial Data** on the Web Best Practices

- Leverages **OpenAPI**

- Focus on **developer experience and usability**

- Modular building blocks **for access to spatial data that can be used in data APIs**

- **Open development**; Public GitHub, Early implementations, In-depth validation

Modular APIs

- A **common core** that is recommended for all implementations of OGC API Standards.

- Common descriptive resources which allow clients to learn the purpose and capabilities of an API as well as how they should be used. These resources are defined in the **Landing Page Requirements Class**

- Clear **separation** between common requirements and more resource specific capabilities.

- The **OGC API - Common** Standard specifies the *common* **requirements** that may be relevant to almost anyone who wants to build an API for spatial resources.

- **Resource-specific requirements** are addressed in resource-specific OGC Standards.

Modular APIs

- **Technologies that change more frequently** are decoupled and specified in separate modules ("**conformance classes**" in OGC terminology). This enables, for example, the use/re-use of new encodings for spatial data or API descriptions.

- **Modularization is not just about a single "service".** OGC APIs provide building blocks that can be reused in APIs in general. In other words, a server that implements the OGC API – Features Standard should not be seen as a standalone service. Rather, this server should be viewed as a collection of API building blocks that together implement the capabilities that are specified in OGC API - Features.

OGC API standards

- OGC API – Common
- OGC API – Features
- OGC API – Coverages
- OGC API – Records
- OGC API – Processes
- OGC API – Tiles
- OGC API – Maps
- OGC API - Environmental Data Retrieval
- OGC API – Styles
- OGC API – Routing

OGC API standards
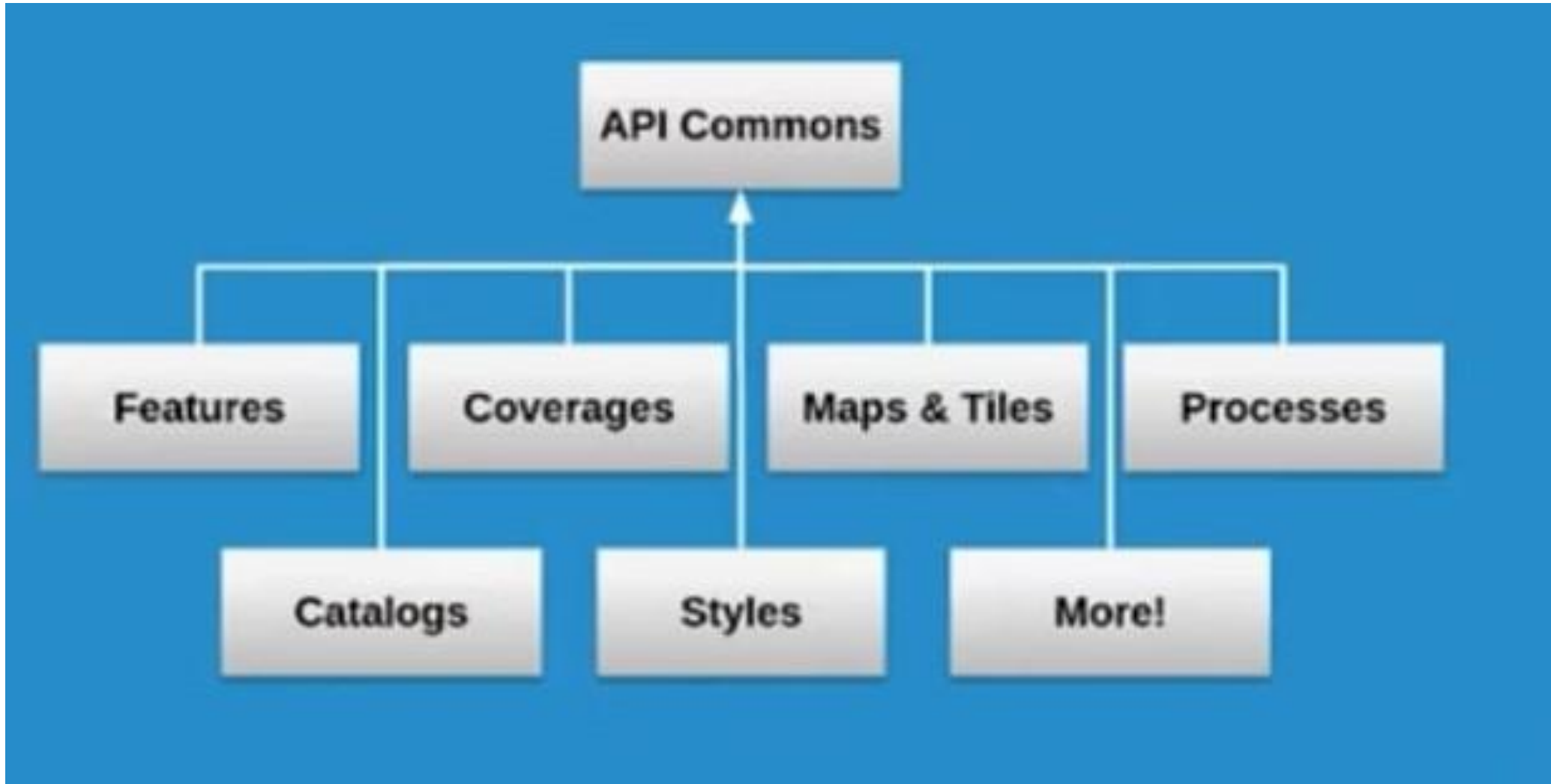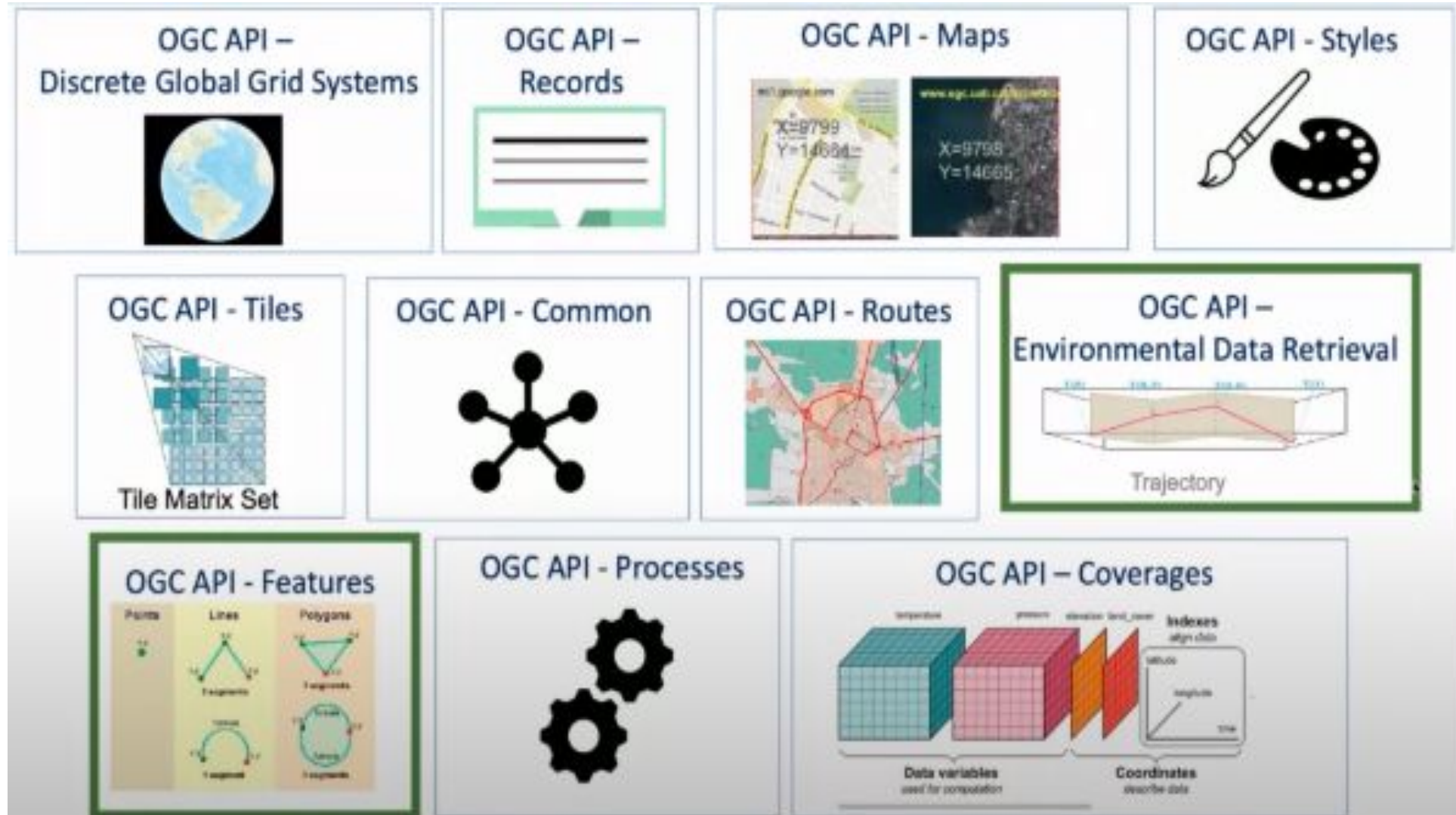
OGC API standards

OGC API standards usage
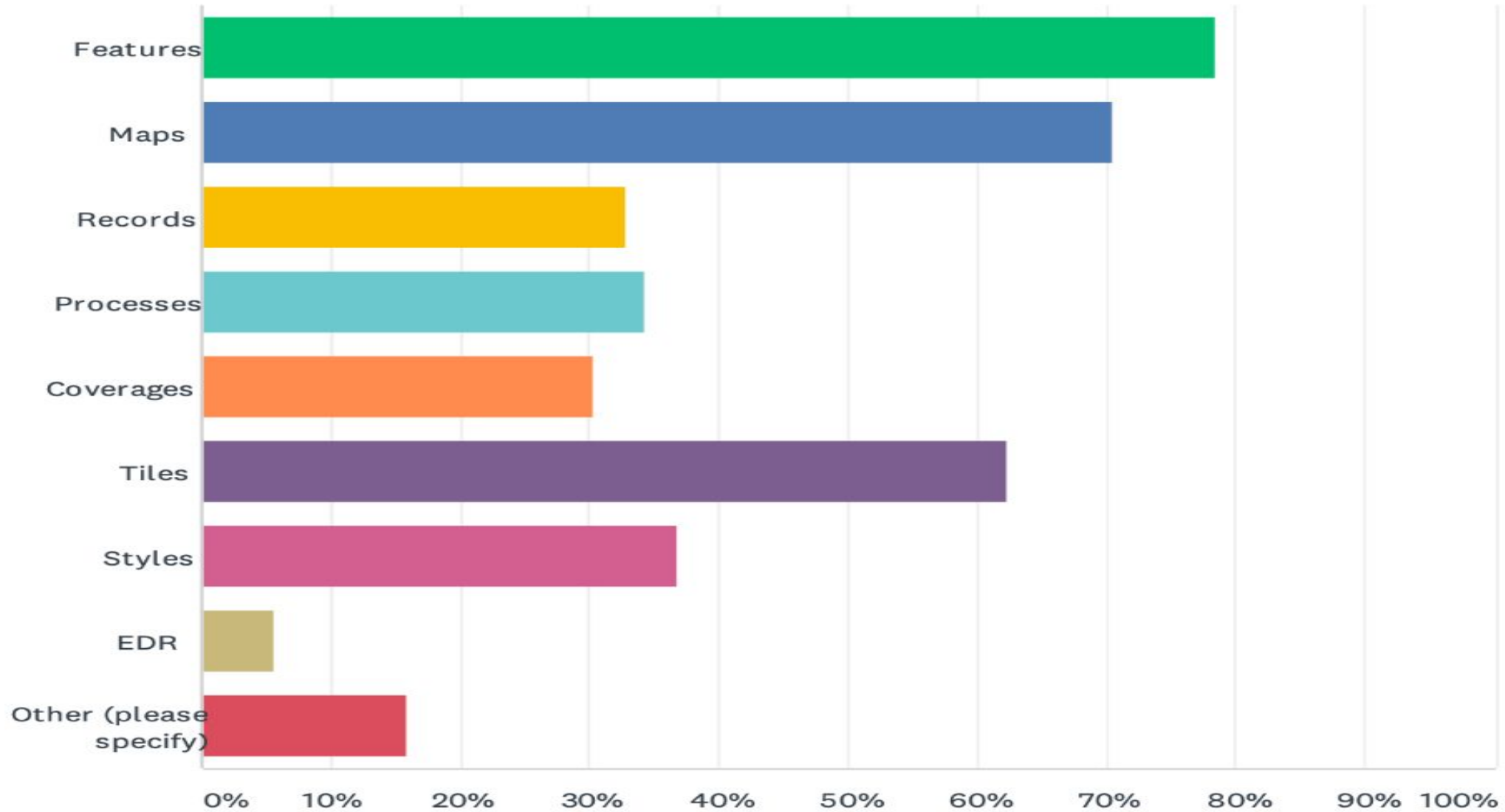
How to approach an OGC API

There are at least two ways to approach an OGC API:

• **Read the landing page, look for links, follow them and discover new links** until the desired resource is found

• **Read a Web API definition document** that will specify a list of paths and path templates to resources

How to approach an OGC API

Resource name	Common path
Landing page[4]	{datasetRoot}/
Conformance declaration[4]	{datasetRoot}/conformance
Dataset Maps	
Dataset maps in the default style [1]	{datasetRoot}/map
Dataset maps[1,2]	{datasetRoot}/styles/{styleId}/map
Dataset map tiles[1,3]	{datasetRoot}/map/tiles/{tileMatrixSetId}/···
Geospatial data collections[5]	
Collections[5]	{datasetRoot}/collections
Collection[5]	{datasetRoot}/collections/{collectionId}
Collection maps in the default style	{datasetRoot}/collections/{collectionId}/map
Collection maps[2]	{datasetRoot}/collections/{collectionId}/styles/{styleId}/map
Collection map tiles[3]	{datasetRoot}/collections/{collectionId}/map/tiles/{tileMatrixSetId}/···

How to approach an OGC API
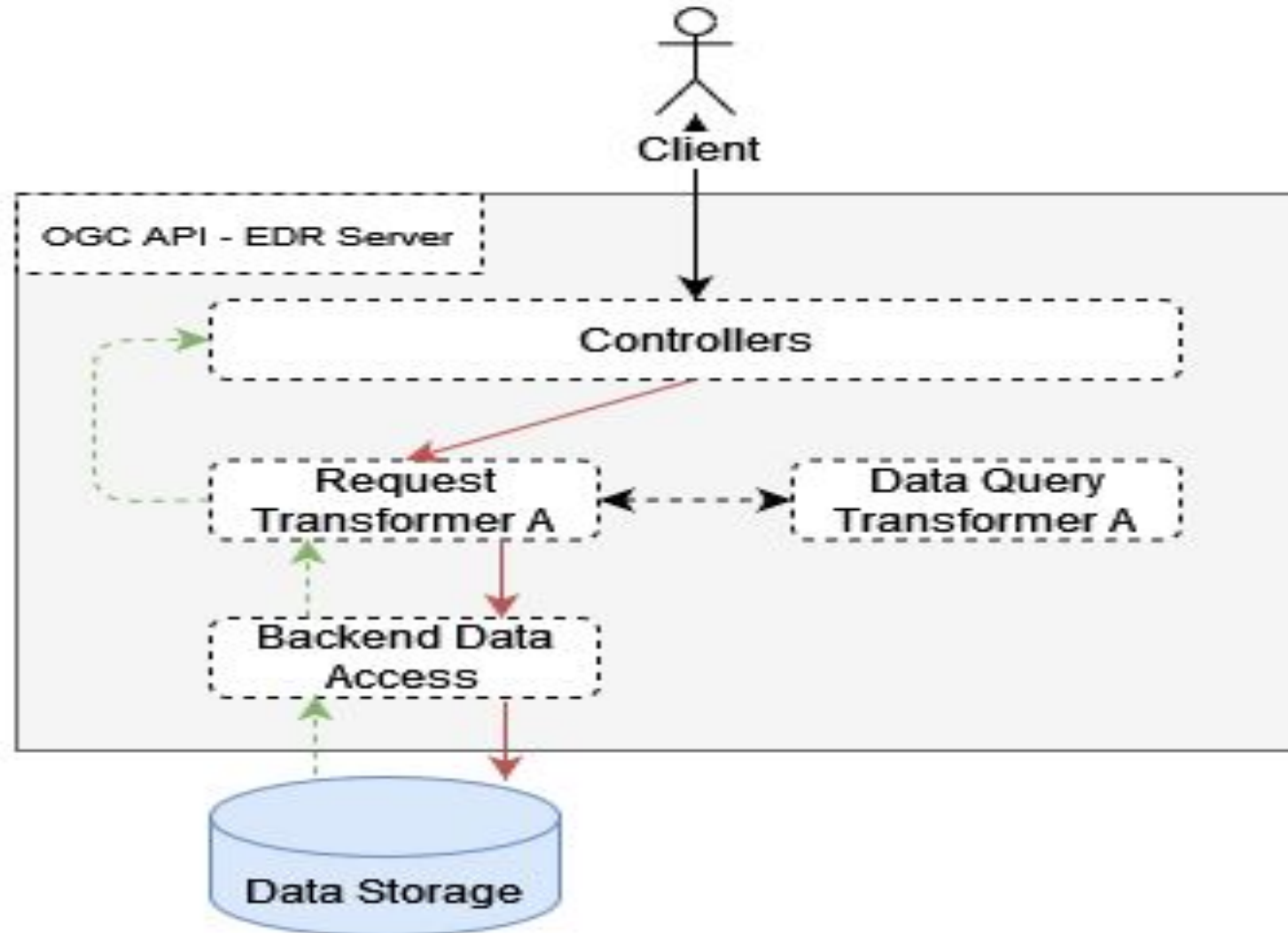
How to approach an OGC API
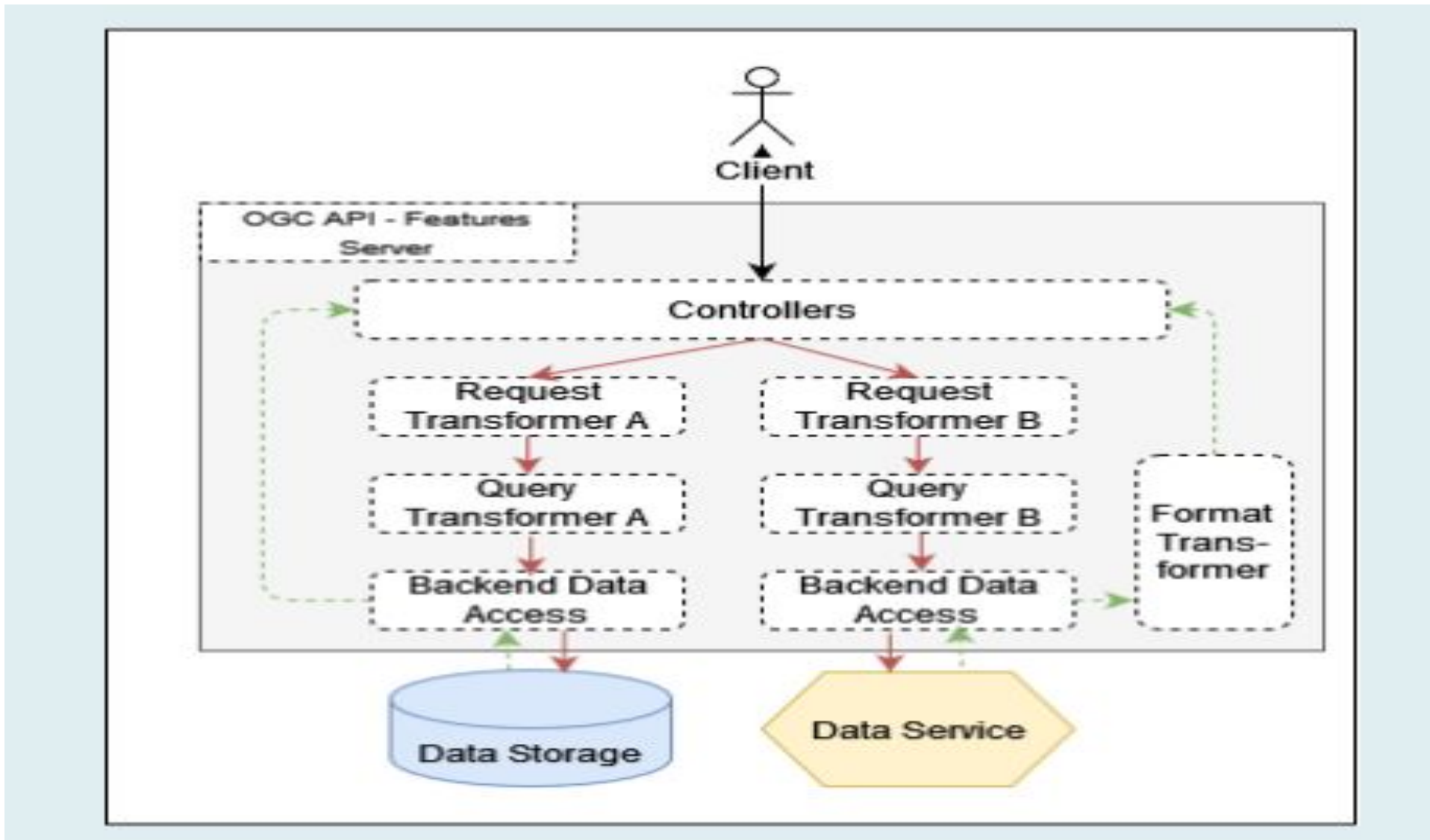
Developers from the geospatial community
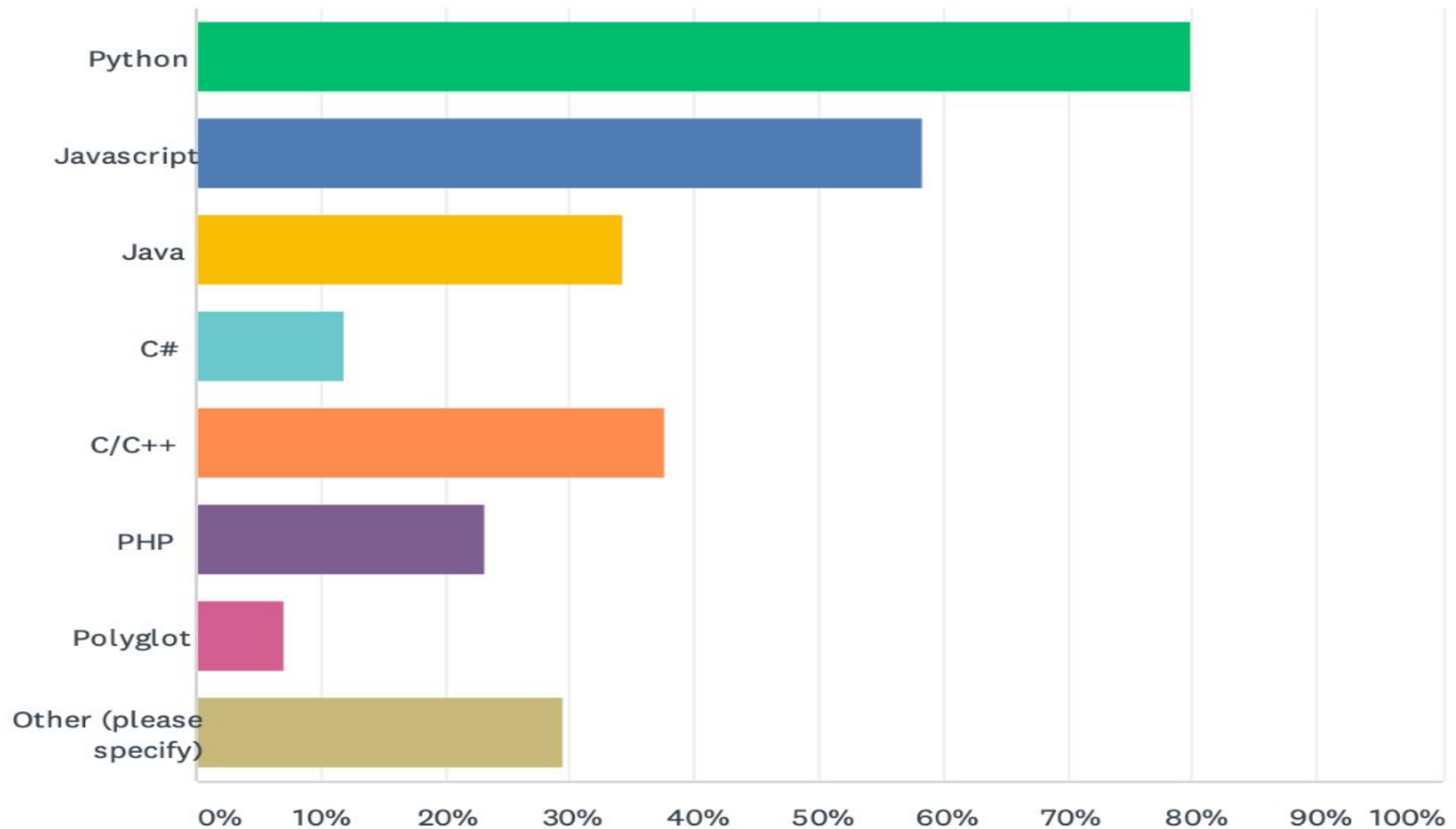
Developers from the geospatial community
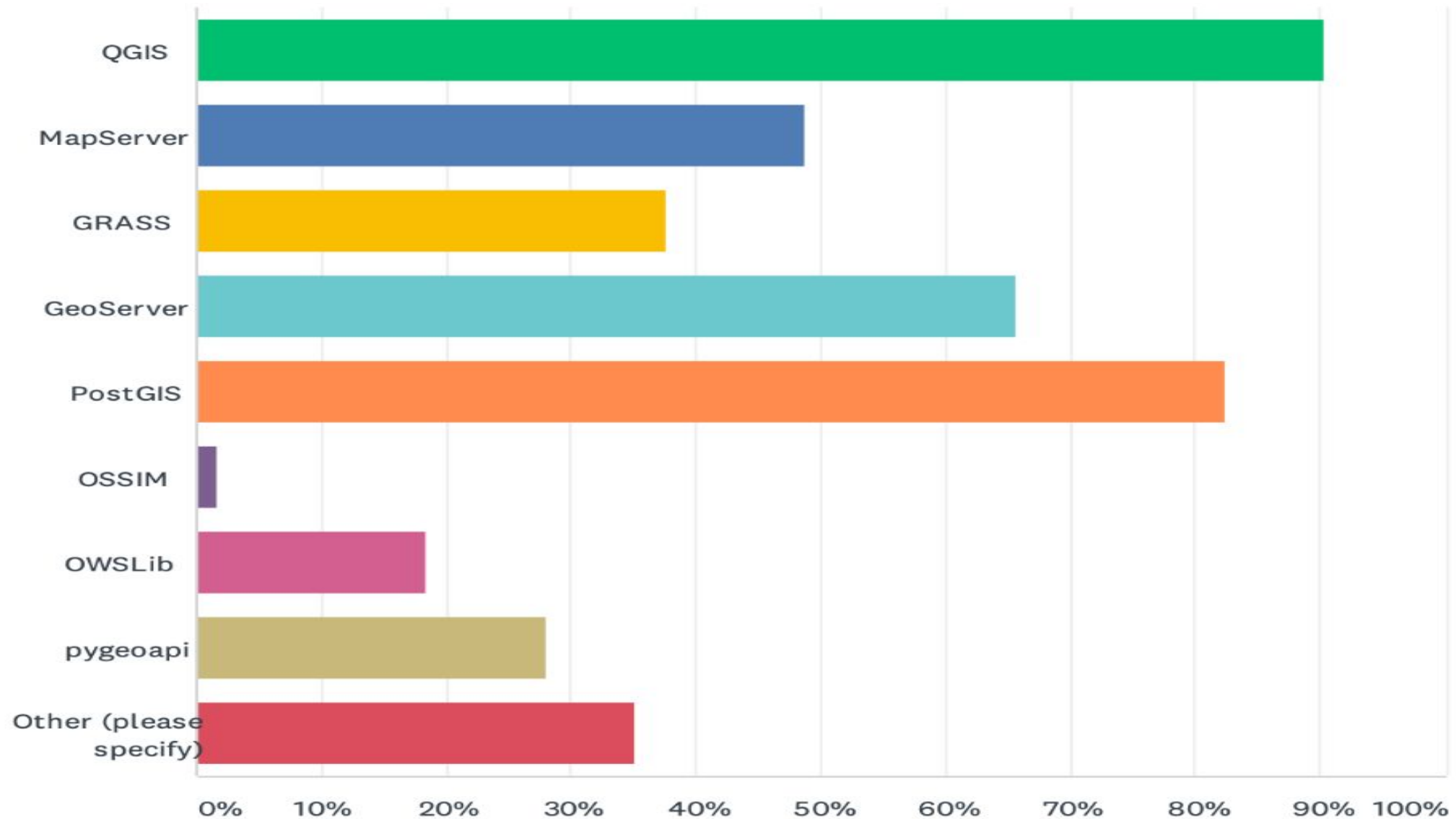
OGC Members

Organization	Level	Region
European Space Agency (ESA)	Strategic	Europe
GeoConnections - Natural Resources Canada	Strategic	North America
Ordnance Survey	Strategic	Europe
United Kingdom Hydrographic Office (UKHO)	Strategic	Europe
US Department of Homeland Security (DHS)	Strategic	North America
US Geological Survey (USGS)	Strategic	North America
US National Aeronautics and Space Administration (NASA)	Strategic	North America
US National Geospatial-Intelligence Agency (NGA)	Strategic	North America
US National Oceanic and Atmospheric Administration (NOAA)	Strategic	North America

Domains that use and develop OGC standards

- Aviation
- Built Environment & 3D
- Business Intelligence
- Defence & Intelligence (D&I)
- Emergency Response & Disaster Management
- Energy & Utilities
- Geosciences & Environment
- Government & Spatial Data Infrastructure
- Mobile Internet & Location Services
- Sensor Webs
- University and Research

Domains that use and develop OGC standards
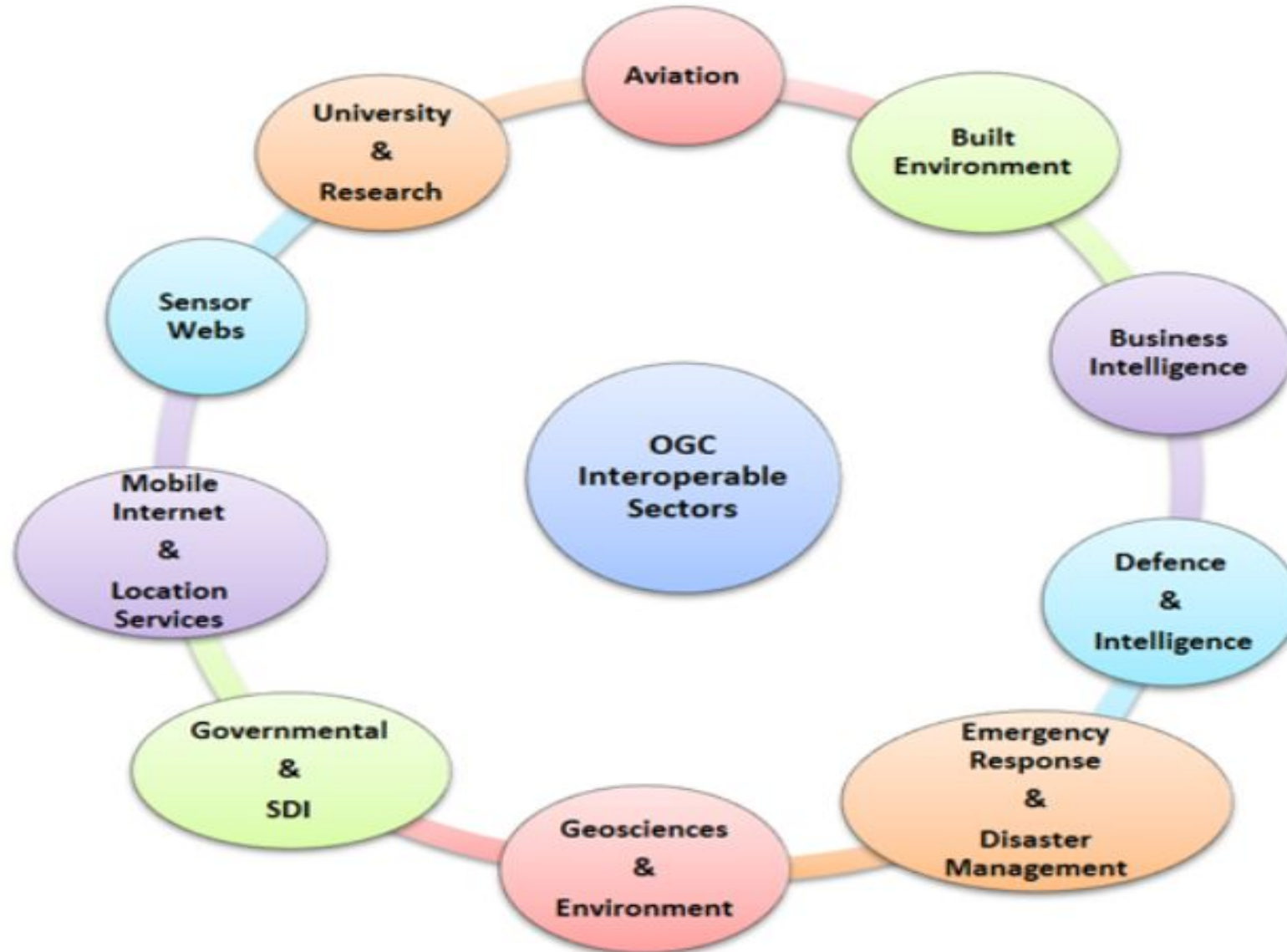
Hands-on: Ordnance Survey

Hands-on: Login to OS Data

Hands-on: Dashboard

Hands-on: Create Project

Hands-on: Create Project

THANK YOU!

prajwalita@iitb.ac.in
prajwalita.chavan@gmail.com

#OGCAPI