GISE HUB | Geospatial Information Science & Engineering

OGC Stack Winter School

15–31 Dec, 2022

Venue: IIIT Hyderabad

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

भारतीय प्रौद्योगिकी संस्थान मुंबई INDIAN INSTITUTE OF TECHNOLOGY BOMBAY
ज्ञानम् परमम् ध्येयम्

IIT Tirupati
Navavishkār
I-Hub Foundation

Today

0930-1000	Recap	Participants		
1000-1130	Graph DB walkthough	Sumit Sen	Slides from Sumit	GeoSPARQL
1130-1145	Tea			
1145-1300	HDF5	Sumit Sen	Slides from Sumit	GitHub - HDFGroup/hdf5: Official HDF5® Library Repository
1300-1400	Lunch	-	-	
1400-1530	Zarr	Sumit Sen	Slides from Sumit	
1530-1545	Tea			
1545-1700	CDB	Sumit Sen	Slides from Sumit	

GraphDB

- A graph database (GDB) is a database that uses graph structures for semantic queries with ***nodes, edges, and properties to represent and store data***.
- Key concept of the system is the graph (or edge or relationship). The graph relates the data items in the store to a ***collection of nodes and edges***, the edges representing the relationships between the nodes.
- Graph databases hold the ***relationships between data as a priority***.
- Querying relationships is fast because they are perpetually stored in the database.
- **Relationships** can be intuitively **visualized** using graph databases, making them useful for ***heavily inter-connected data***

Types of GraphDBs

Amazon Neptune
Purpose-built graph database that stores billions of relationships

ORACLE DATABASE 12c
Oracle Spatial and Graph
ORACLE

neo4j

redis

Microsoft Graph

JanusGraph

OPENLINK VIRTUOSO

TigerGraph

TerminusDB

teradata.

TIBCO®

GraphDB

MarkLogic

SPARQL and GeoSPARQL

- SPARQL Protocol and RDF Query Language) is an RDF query language
- SPARQL allows users to write queries against what can loosely be called "key-value" data or, more specifically, data that follow the RDF specification of the W3C.
- Thus, the entire database is a set of "subject-predicate-object" triples.
- This is analogous to some NoSQL databases' usage of the term "document-key-value“
- GeoSPARQL extends Geospatial support to SPARQL

Some Use Cases

- Existing LOD RDF datasets with some simple spatial data
  - DBPedia, GeoNames, etc.
  - Mainly simple, WGS84 Lat Long points
- GIS applications with semantically complex thematic aspects
  - Reasoning based on complex land cover type hierarchy
- Qualitative reasoning applications without explicit geometries
  - RCC8-based reasoning
- Data integration applications
  - RDB2RDF + Federated SPARQL over spatial datasets

Goals for GeoSPARQL

- Provide a common target for implementers & users
  - Oracle, BBN Parliament, Virtuoso, OWLIM, Franz, …
- Work within SPARQL's extensibility framework
- Simple enough for general users but capable enough for GIS professionals
- Accommodate systems based on qualitative spatial reasoning and systems based on quantitative geometries
- Don't re-invent the wheel!

I
ISO 13249 – SQL/MM

Simple Features
Well Known Text (WKT)
GML
KML
GeoJSON

A Simple Example

Example Data: OGCStackWinterSchool located in Hyderabad

```
:OGCStackWinterSchool  :locatedIn  :Hyderabad .
:Hyderabad      :hasPointGeometry [
    ogc:asWKT “POINT(78.43 17.24)”^^ogc:WKTLiteral ] .
```

Example Query: Find airports within 100 KM of IIITHyderabad

```
SELECT ?airport
WHERE { ?airport rdf:type :Airport .
        ?airport :hasPointGeometry [
                 ogc:asWKT ?aPointGeom ]
        FILTER(ogcf:distance(?aPointGeom,
                 “POINT(78.43 17.24)”^^ogc:WKTLiteral,
                  ogc:km) <= 100) }
```

What Does GeoSPARQL Give Us?

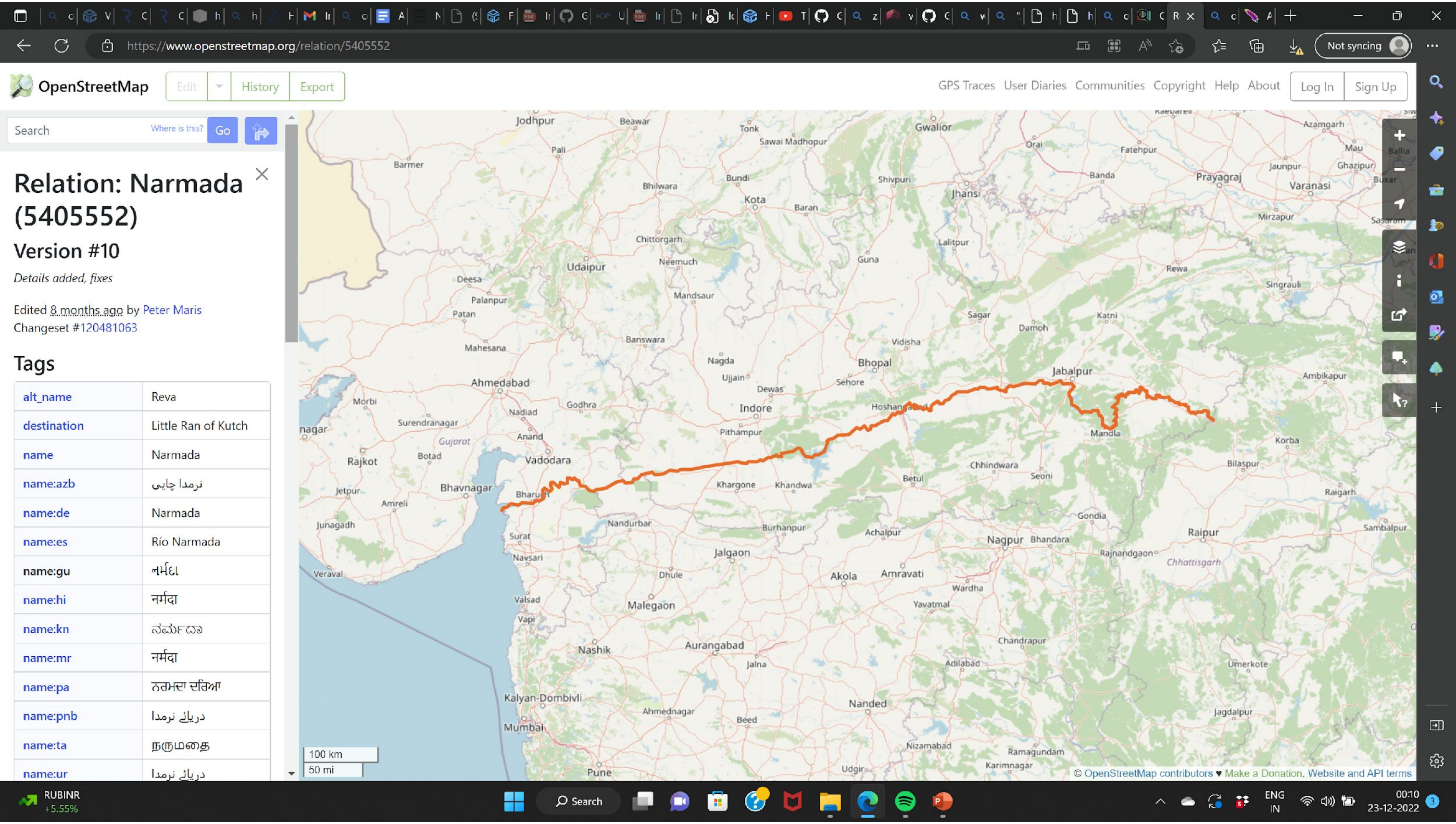
- Vocabulary for Query Patterns
  - Classes
    - Spatial Object, Feature, Geometry
  - Properties
    - Topological relations
    - Links between features and geometries
  - Datatypes for geometry literals
    - ogc:WKTLiteral, ogc:GMLLiteral
- Query Functions
  - Topological relations, distance, buffer, intersection, …
- Entailment Components
  - RIF rules to expand feature-feature query into geometry query
  - Gives a common interface for qualitative and quantitative systems

Some Features of the Spec

- Uses a Modular Design
  - Consists of several components
- It’s Parameterized
  - *Serialization*: WKT, GML, KML, …
  - *Relation Family*: Simple Features, Egenhofer, RCC8, …

- Benefits
  - Vendors can easily identify what they support
    - We support components X, Y and Z for WKT and GML serializations and Simple Features relations
  - Extensible
    - Can easily add new components
    - Can easily add new spatial relations and serializations

Visualizing an example



Relation: Narmada (5405552)

Version #10

Details added, fixes

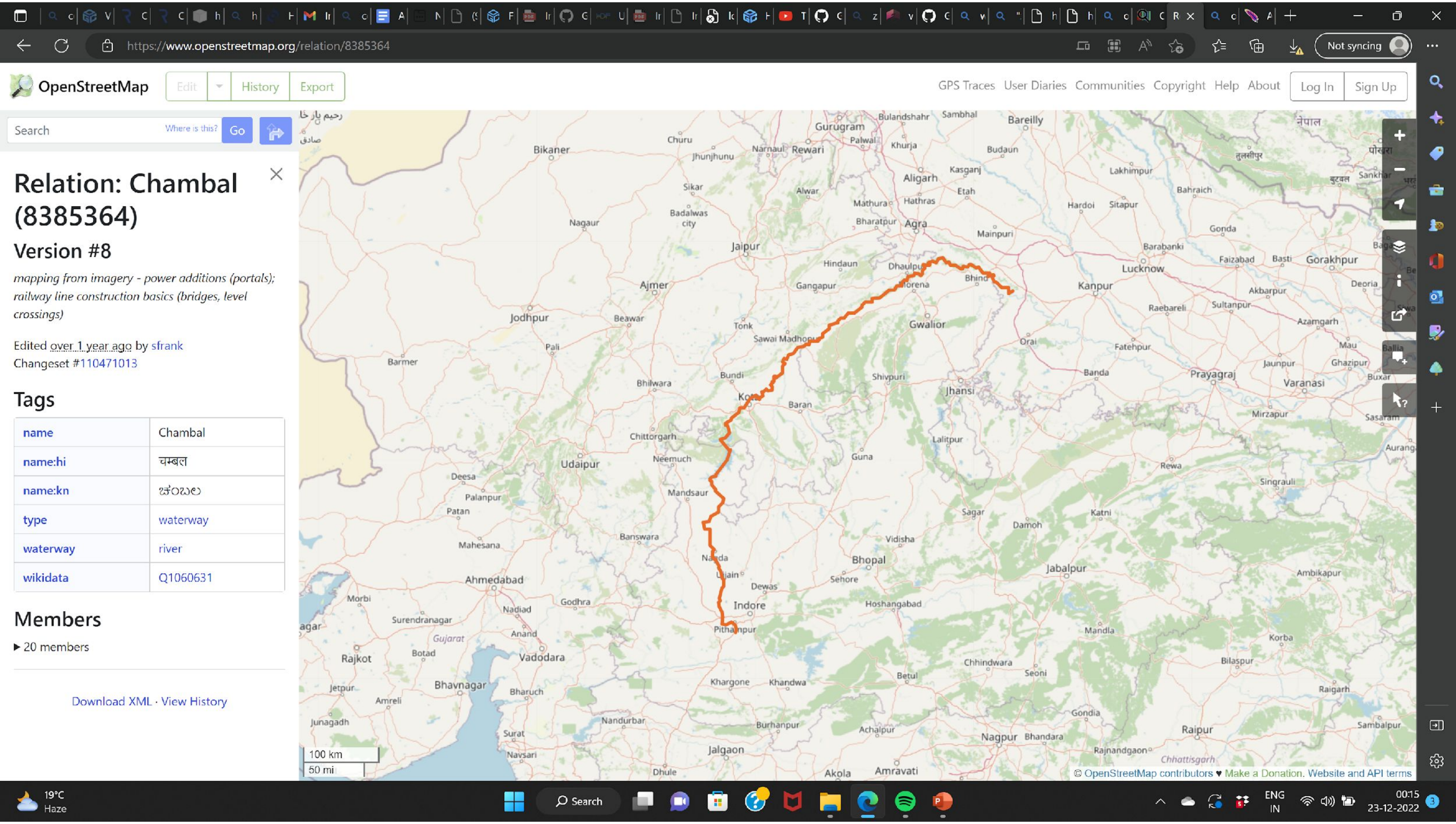
Edited 8 months ago by Peter Maris
Changeset #120481063

Tags

alt_name	Reva
destination	Little Ran of Kutch
name	Narmada
name:azb	نرمدا چایی
name:de	Narmada
name:es	Río Narmada
name:gu	નર્મદા
name:hi	नर्मदा
name:kn	ನರ್ಮದಾ
name:mr	नर्मदा
name:pa	ਨਰਮਦਾ ਦਰਿਆ
name:pnb	دریائے نرمدا
name:ta	நருமதை
name:ur	دریائے نرمدا

https://www.openstreetmap.org/relation/5405552

Another example



Relation: Chambal (8385364)

Version #8

mapping from imagery - power additions (portals); railway line construction basics (bridges, level crossings)

Edited over 1 year ago by sfrank
Changeset #110471013

Tags

name	Chambal
name:hi	चम्बल
name:kn	ಚಂಬಲ
type	waterway
waterway	river
wikidata	Q1060631

Members

▶ 20 members

Download XML · View History

https://www.openstreetmap.org/relation/8385364

Example Data

Meta Information

```
:City          rdfs:subClassOf    ogc:Feature .
:Park      rdfs:subClassOf    ogc:Feature .
:exactGeometry rdfs:subPropertyOf ogc:hasGeometry .
```

Non-spatial Properties

```
:Nashua        rdf:type    :City .
:MinesFallsParkrdf:type    :Park .
:MinesFallsPark:opened     “1950-03-01”^^xsd:date .
```

Spatial Properties

```
:IIITCampus :exactGeometry  :geo1 .
:geo1       ogc:asWKT   “Polygon((…))”^^ogc:WKTLiteral .

:Hyderabad      :exactGeometry  :geo2 .
:geo2       ogc:asWKT   “Polygon((…))”^^ogc:WKTLiteral .

:IIITCampus ogc:sfWithin    :Hyderabad .
```

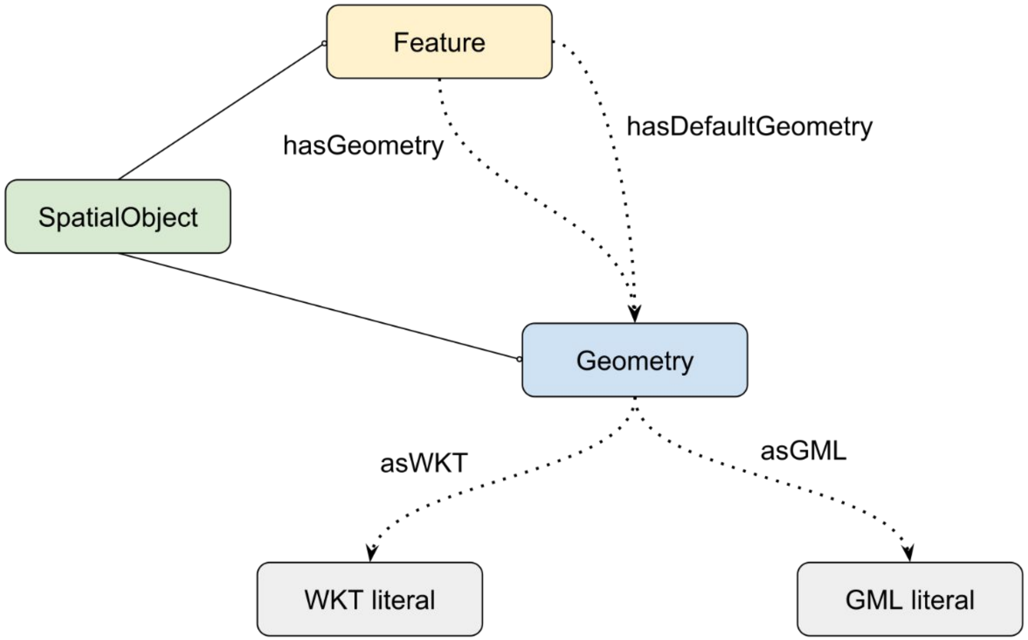
Try RDF dumps

- Index of /results/osm-to-rdf/poi/asia (slipo.eu)

- OpenStreetMap TTL Extracts (uni-freiburg.de)

There are different formats

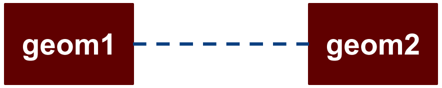
- Turtle, RDFs,



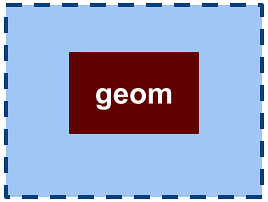
GeoSPARQL Query Functions

GeoSPARQL Query Functions

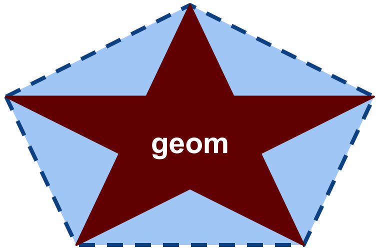
- `ogcf:distance(geom1: ogc:WKTLiteral, geom2: ogc:WKTLiteral, units: xsd:anyURI): xsd:double`



- `ogcf:buffer(geom: ogc:WKTLiteral, radius: xsd:double, units: xsd:anyURI): ogc:WKTLiteral`

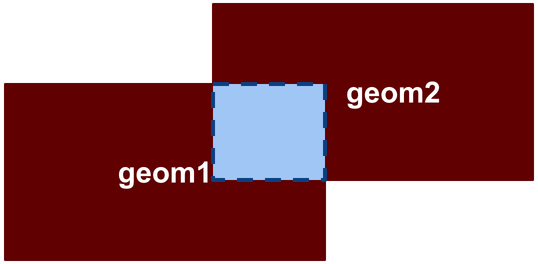


- `ogcf:convexHull(geom: ogc:WKTLiteral): ogc:WKTLiteral`

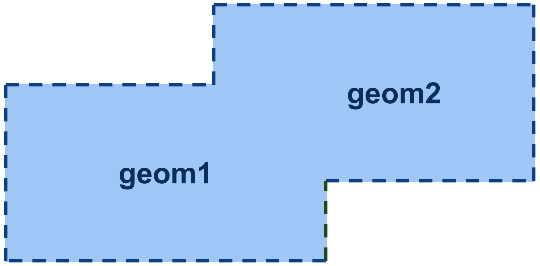


GeoSPARQL Query Functions

```
– ogcf:intersection(geom1: ogc:WKTLiteral,
                    geom2: ogc:WKTLiteral): ogc:WKTLiteral
```

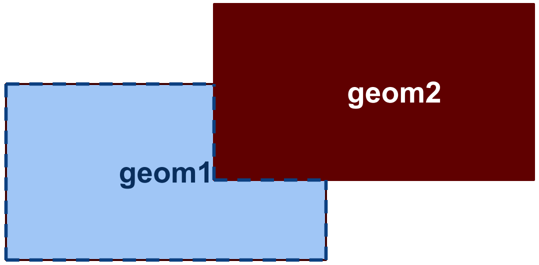


```
– ogcf:union(geom1: ogc:WKTLiteral,
             geom2: ogc:WKTLiteral): ogc:WKTLiteral
```

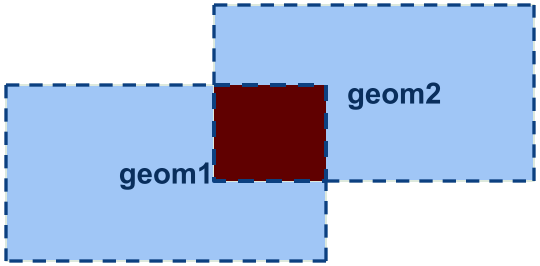


GeoSPARQL Query Functions

```
– ogcf:difference(geom1: ogc:WKTLiteral,
                  geom2: ogc:WKTLiteral): ogc:WKTLiteral
```

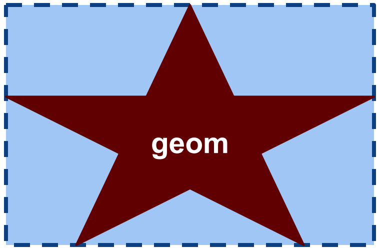


```
– ogcf:symDifference(geom1: ogc:WKTLiteral,
                     geom2: ogc:WKTLiteral): ogc:WKTLiteral
```

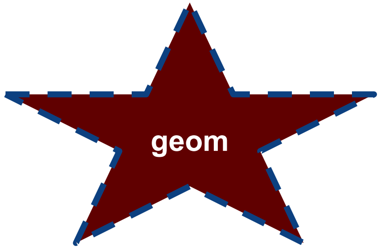


GeoSPARQL Query Functions

- `ogcf:envelope(geom: ogc:WKTLiteral): ogc:WKTLiteral`



- `ogcf:boundary(geom1: ogc:WKTLiteral): ogc:WKTLiteral`



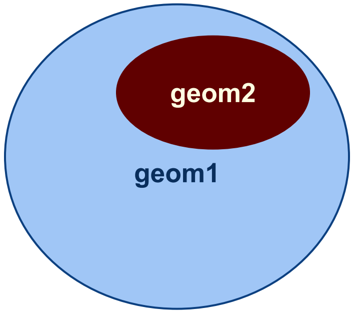
- `ogcf:getSRID(geom: ogc:WKTLiteral): xsd:anyURI`

GeoSPARQL Topological Query Functions

```
– ogcf:relate(geom1: ogc:WKTLiteral,
              geom2: ogc:WKTLiteral,
              patternMatrix: xsd:string): xsd:boolean
```

DE-9IM Intersection Matrix

		geom2		
		Interior	Boundary	Exterior
geom1	Interior	T	T	T
	Boundary	F	F	T
	Exterior	F	F	T



ogc:contains

patternMatrix: TTTFFTFFT

GeoSPARQL Topological Query Functions

```
– ogcf:sfEquals(geom1: ogc:WKTLiteral,
                geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfDisjoint(geom1: ogc:WKTLiteral,
                  geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfIntersects(geom1: ogc:WKTLiteral,
                    geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfTouches(geom1: ogc:WKTLiteral,
                 geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfCrosses(geom1: ogc:WKTLiteral,
                 geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfWithin(geom1: ogc:WKTLiteral,
                geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfContains(geom1: ogc:WKTLiteral,
                  geom2: ogc:WKTLiteral): xsd:boolean
– ogcf:sfOverlaps(geom1: ogc:WKTLiteral,
                  geom2: ogc:WKTLiteral): xsd:boolean
```

Assumes Simple Features Relation Family

Non-topological Query Functions

```
– ogcf:distance(geom1: ogc:WKTLiteral, geom2: ogc:WKTLiteral,
                units: xsd:anyURI): xsd:double
– ogcf:buffer(geom: ogc:WKTLiteral, radius: xsd:double,
              units: xsd:anyURI): ogc:WKTLiteral
– ogcf:convexHull(geom: ogc:WKTLiteral): ogc:WKTLiteral
– ogcf:intersection(geom1: ogc:WKTLiteral,
                    geom2: ogc:WKTLiteral): ogcf:WKTLiteral
– ogcf:union(geom1: ogc:WKTLiteral,
             geom2: ogc:WKTLiteral): ogc:WKTLiteral
– ogcf:difference(geom1: ogc:WKTLiteral,
                  geom2: ogc:WKTLiteral): ogcf:WKTLiteral

– ogcf:symDifference(geom1: ogc:WKTLiteral,
                     geom2: ogc:WKTLiteral): ogc:WKTLiteral
– ogcf:envelope(geom: ogc:WKTLiteral): ogcf:WKTLiteral

– ogcf:boundary(geom1: ogc:WKTLiteral): ogc:WKTLiteral
```

Some Examples to try

- Add data

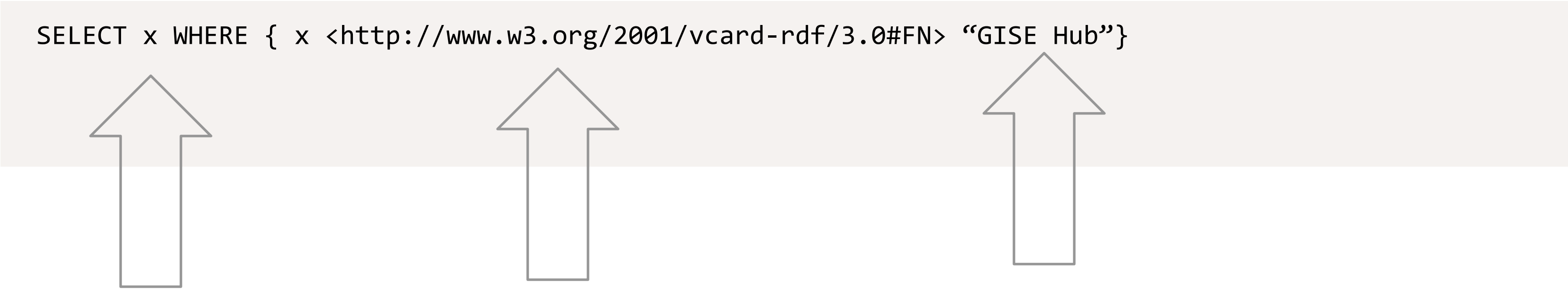
```
@prefix geo: <http://www.opengis.net/ont/geosparql#> .
@prefix sf: <http://www.opengis.net/ont/sf> .
@prefix ex: <http://example.org/> .

ex:HydAirport a ex:Landmark ;
          geo:hasGeometry ex:coordinates-et.
ex:coordinates-et a sf:Point;
        geo:asWKT "POINT(78.429382 17.240263)"^^geo:wktLiteral .
ex:DelhiAirport a ex:Landmark ;
          geo:hasGeometry ex:coordinates-tb.
ex:coordinates-tb a sf:Point;
        geo:asWKT "POINT(77.095016 28.581039)"^^geo:wktLiteral .
```

- Query

```
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX uom: <http://www.opengis.net/def/uom/OGC/1.0/>
PREFIX ex: <http://example.org/>
SELECT *
WHERE {
  ?lmA a ex:Landmark ;
       geo:hasGeometry [ geo:asWKT ?coord1 ].
  ?lmB a ex:Landmark ;
       geo:hasGeometry [ geo:asWKT ?coord2 ].
  BIND((geof:distance(?coord1, ?coord2, uom:metre)/1000) as ?dist) .
  FILTER (str(?lmA) < str(?lmB))
}
```

Playing with SPARQL, generally

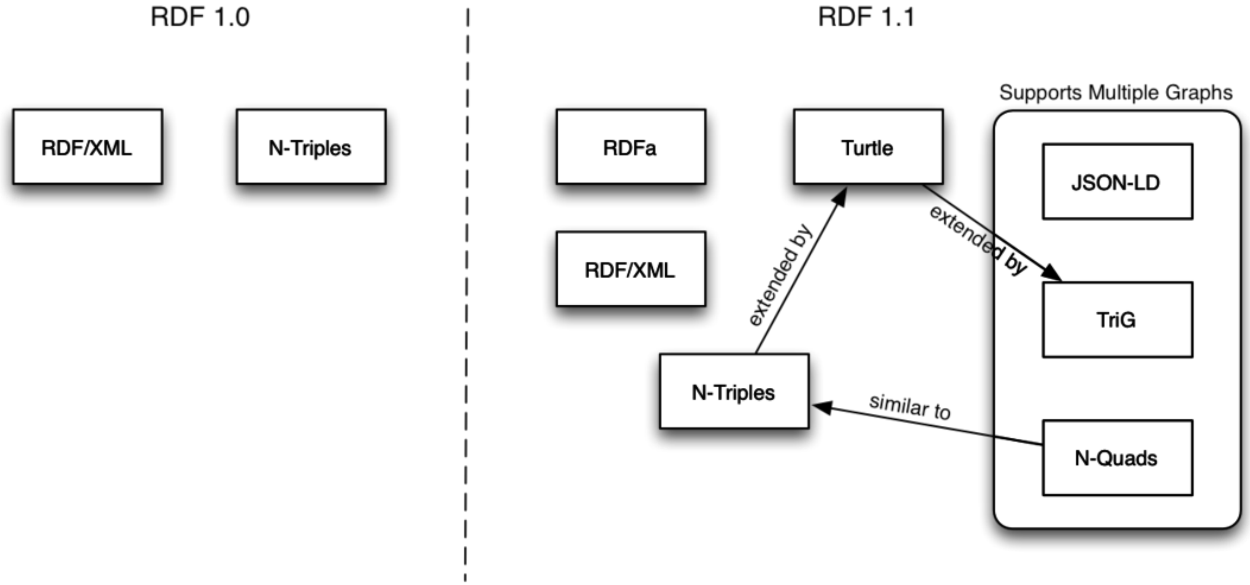
- https://sparql-playground.sib.swiss/



```
SELECT x WHERE { x <http://www.w3.org/2001/vcard-rdf/3.0#FN> “GISE Hub”}
```

From RDF to JSON

- https://www.w3.org/2016/01/json2rdf.html

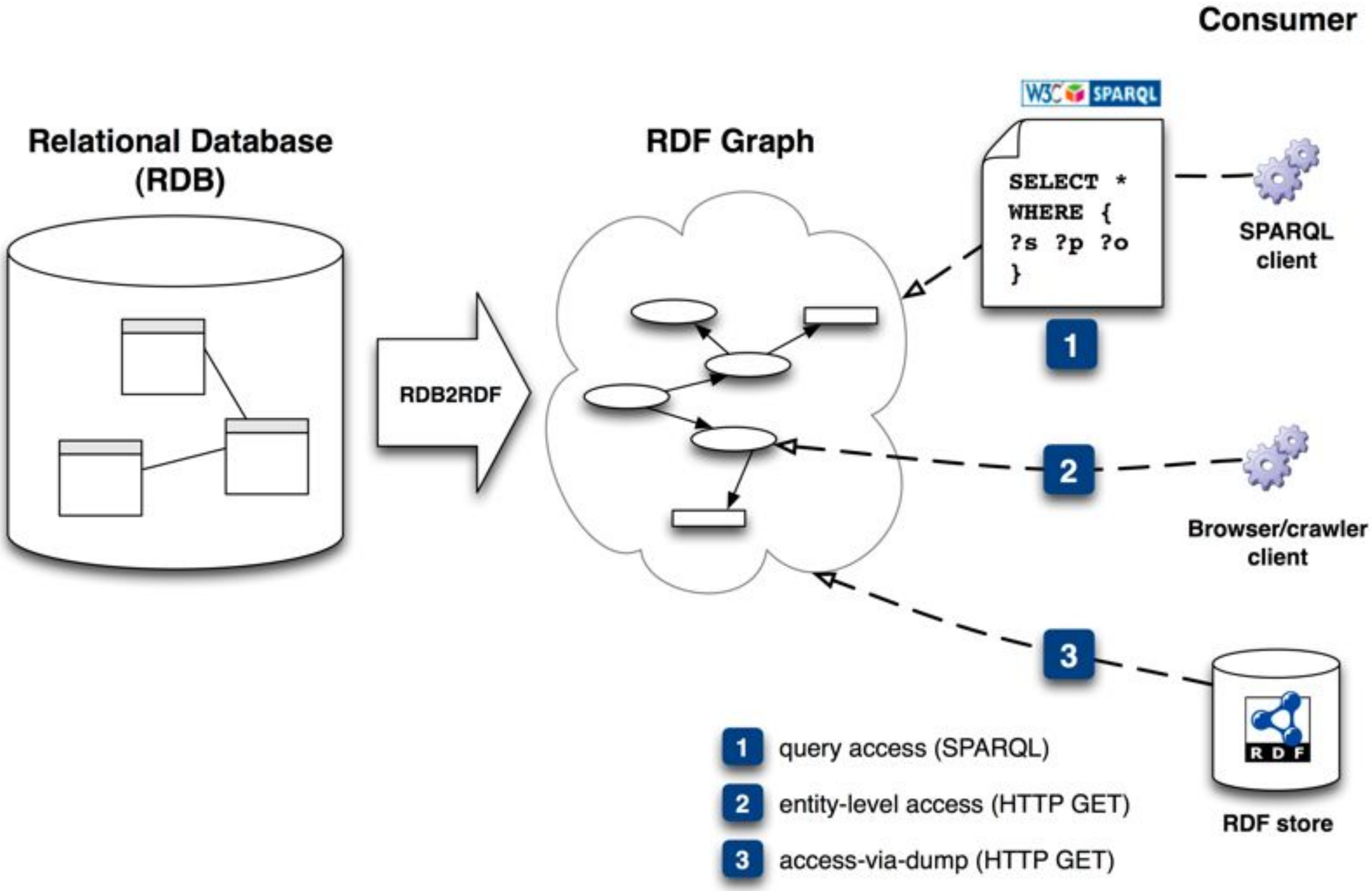


JSON-LD

- Links and semantics for the JSON ecosystem
- Adds a "context" to map JSON to RDF
- RDF → JSON-LD → RDF is lossless

```
{
  "@context": "http://json-ld.org/contexts/person.jsonld",
  "@id": "http://example.org/alice.foaf#me",
  "name": "Alice Hacker",
  "homepage": "http://example.org/alice",
}
```

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

<http://example.org/alice.foaf#me>
       foaf:name  "Alice Hacker" ;
       foaf:homepage <http://example.org/alice> .
```



```
SELECT *
WHERE {
?s ?p ?o
}
```

1 query access (SPARQL)

2 entity-level access (HTTP GET)

3 access-via-dump (HTTP GET)

- **TEA Break**

- Ensure you have RDF4J installed
  - Server
  - Workbench

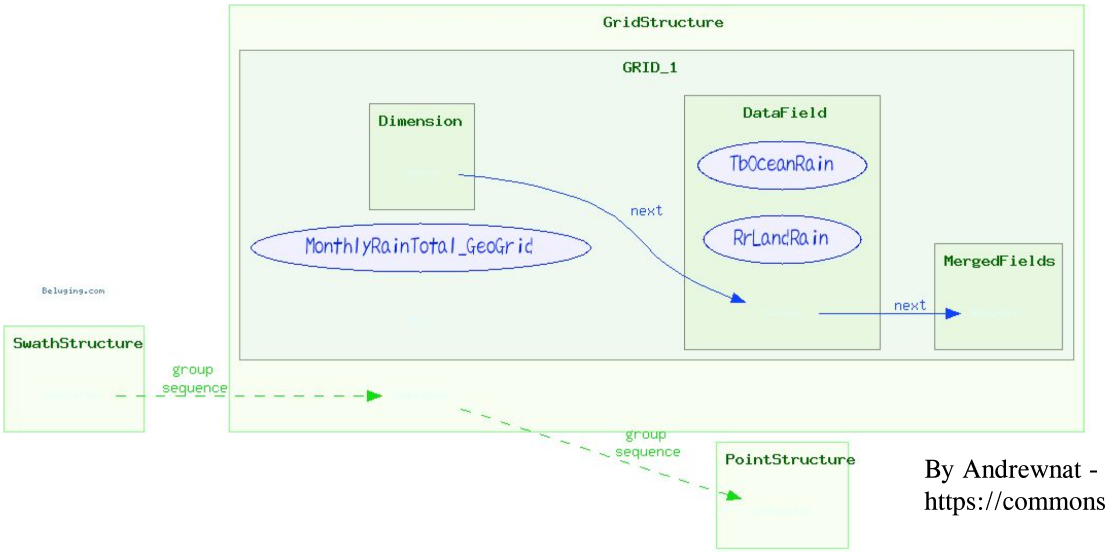
- Try Creating GeoSPARQL data in the workbench

- Try editing the data

- Try Queries

Now on to HDF5

- AEHOO (All Encompassing Hierarchical Object Oriented format) –> Hierarchical Data Format (HDF)
  - HDF is self-describing, allowing an application to interpret the structure and contents of a file with no outside information. One HDF file can hold a mix of related objects which can be accessed as a group or as individual objects. Users can create their own grouping structures called "vgroups."



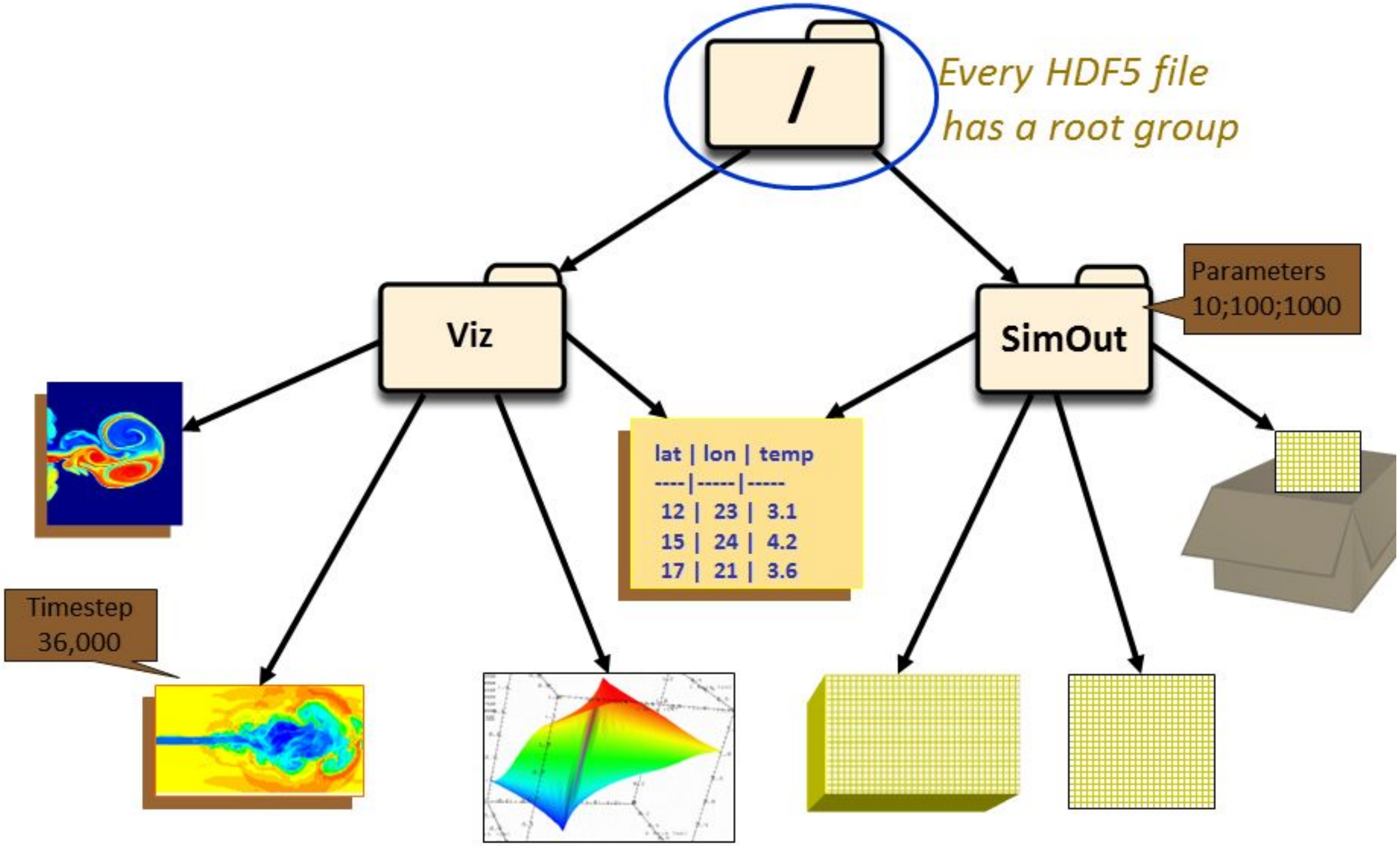
By Andrewnat - https://infchg.appspot.com/t/q5jV, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=10457426

HDF5

- The Hierarchical Data Format version 5 (HDF5), is an open source file format that supports large, complex, heterogeneous data. HDF5 uses a "file directory" like structure that allows you to organize data within the file in many different structured ways, as you might do with files on your computer.
- The HDF5 format also allows for embedding of metadata making it *self-describing*.

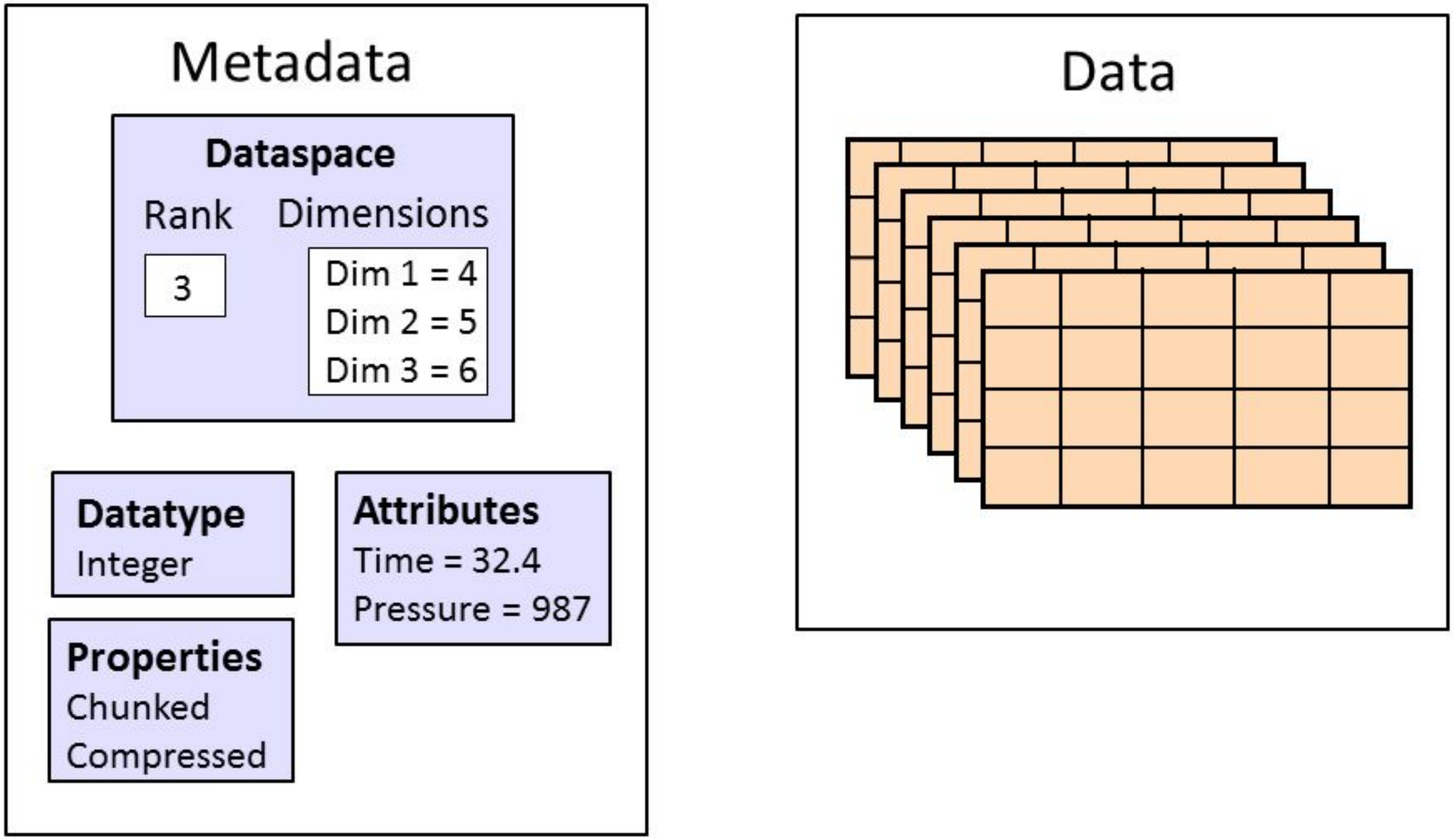
HDF5 is one hierarchical data format, that builds upon both HDF4 and NetCDF (two other hierarchical data formats)

There are two groups in the HDF5 file depicted below: Viz and SimOut. Under the Viz group are a variety of images and a table that is shared with the SimOut group. The SimOut group contains a 3-dimensional array, a 2-dimensional array and a link to a 2-dimensional array in another HDF5 file.



lat	lon	temp
12	23	3.1
15	24	4.2
17	21	3.6

HDF5 datasets organize and contain the “raw” data values. A dataset consists of metadata that describes the data, in addition to the data itself



Metadata

Dataspace

Rank

3

Dimensions

Dim 1 = 4

Dim 2 = 5

Dim 3 = 6

Datatype

Integer

Properties

Chunked

Compressed

Attributes

Time = 32.4

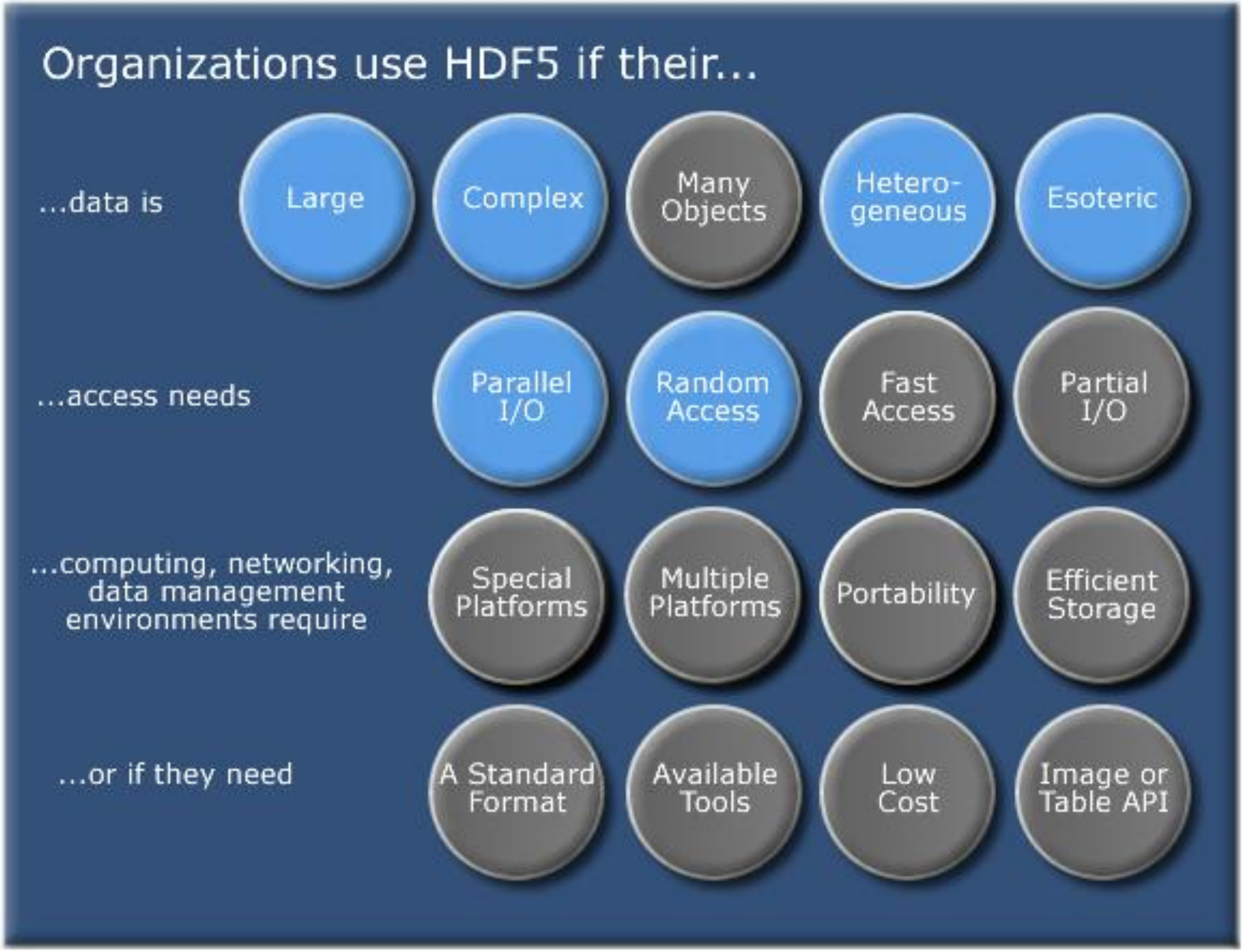
Pressure = 987

Data

Related

- netCDF (NetCDF - NERSC Documentation)
- Zarr (we will see later)
- HDF4 (a previous version)
- Cloud Native
  - A core aim of cloud-native geospatial is to decrease the burden on data providers and in turn enable far more geospatial data to be FAIR. The only cost that providers should need to pay is for the cloud storage, which continues to fall.
- Google Earth Engine (GEE) has been operating on the cloud for years, albeit traditionally as a walled garden (though GEE does now support COG registration)
  - any data on any cloud could be used by anyone

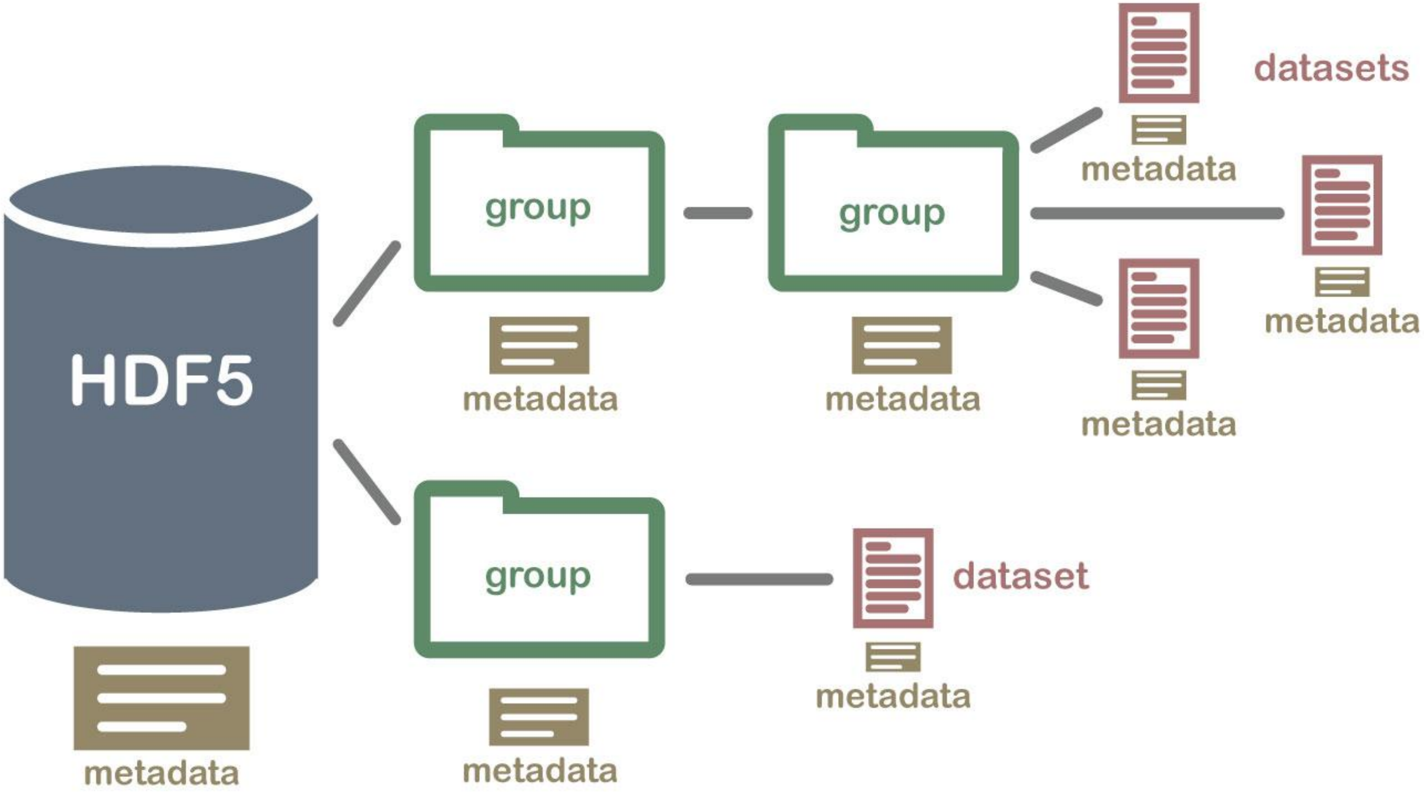
Why use HDF5



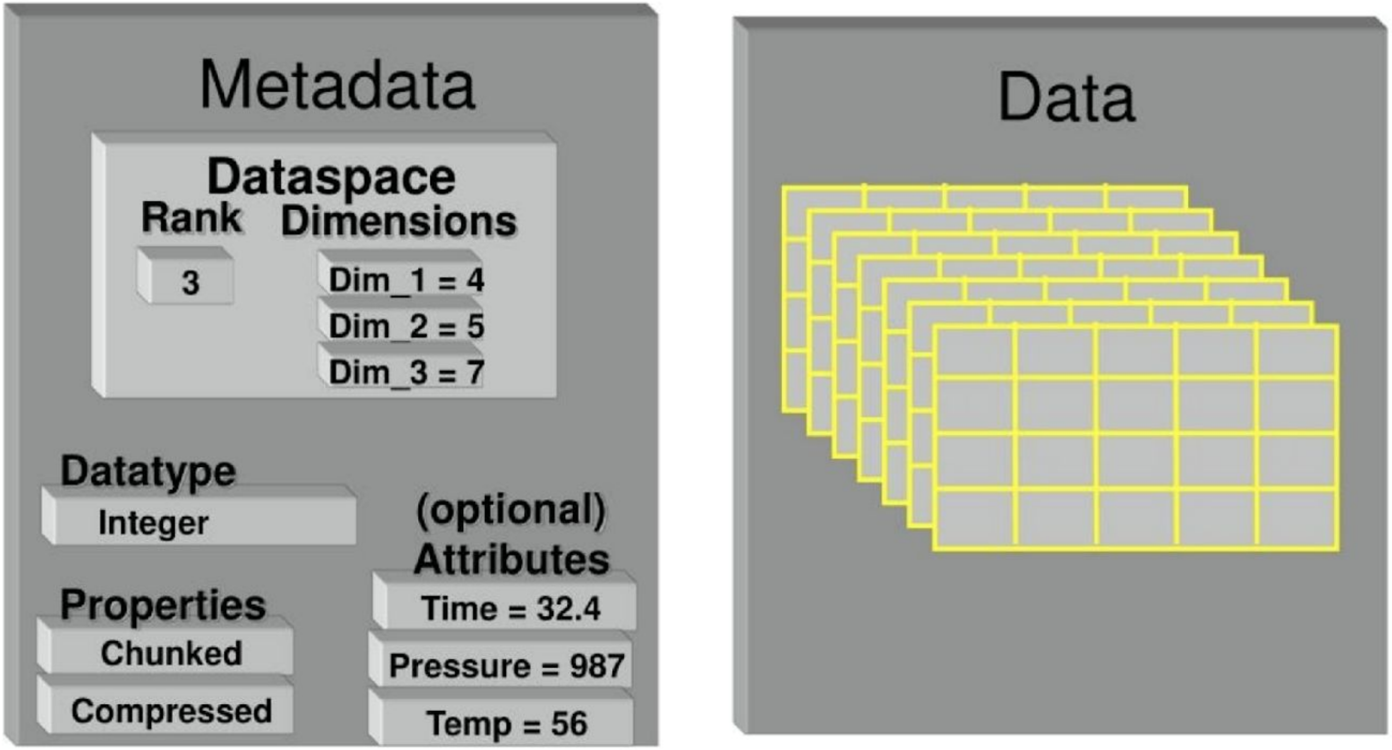
Organizations use HDF5 if their...

...data is	Large	Complex	Many Objects	Hetero-geneous	Esoteric
...access needs		Parallel I/O	Random Access	Fast Access	Partial I/O
...computing, networking, data management environments require		Special Platforms	Multiple Platforms	Portability	Efficient Storage
...or if they need		A Standard Format	Available Tools	Low Cost	Image or Table API

How is HDF5 is structured



HDF5 Dataset



Metadata

Dataspace

Rank

3

Dimensions

Dim_1 = 4

Dim_2 = 5

Dim_3 = 7

Datatype

Integer

Properties

Chunked

Compressed

(optional) Attributes

Time = 32.4

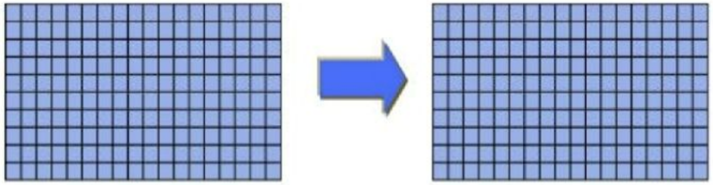
Pressure = 987

Temp = 56

Data

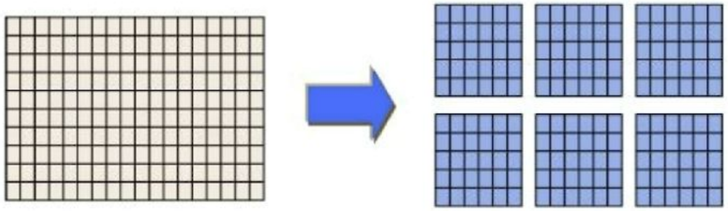
Storage Properties

Contiguous (default)



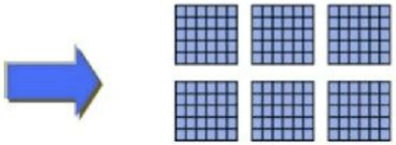
Data elements stored physically adjacent to each other

Chunked



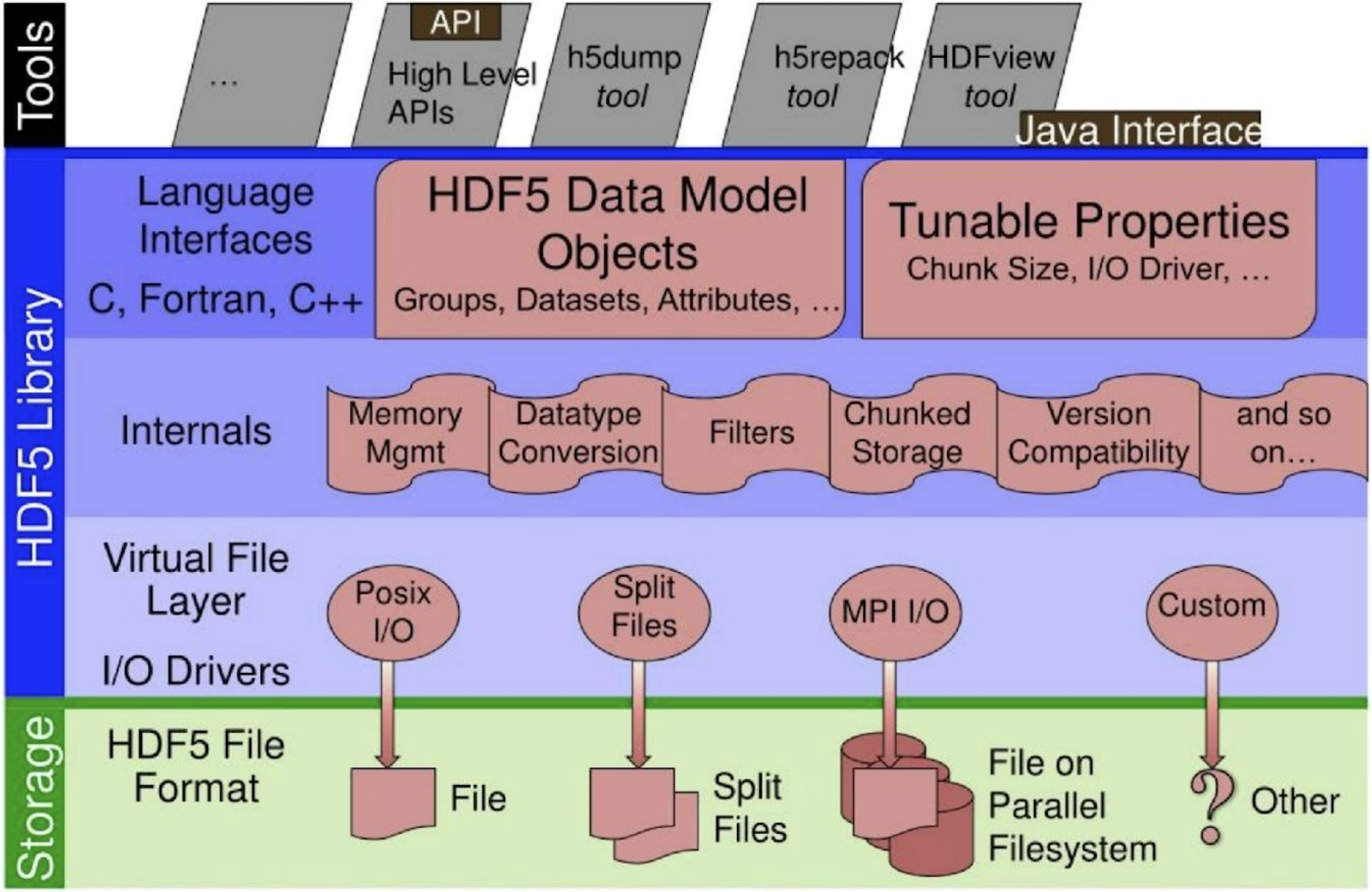
Better access time for subsets; extensible

Chunked & Compressed



Improves storage efficiency, transmission speed

HDF5 Software Layers & Storage



Tools

…

API

High Level APIs

h5dump *tool*

h5repack *tool*

HDFview *tool*

Java Interface

HDF5 Library

Language Interfaces

C, Fortran, C++

HDF5 Data Model Objects

Groups, Datasets, Attributes, …

Tunable Properties

Chunk Size, I/O Driver, …

Internals

Memory Mgmt

Datatype Conversion

Filters

Chunked Storage

Version Compatibility

and so on…

Virtual File Layer

I/O Drivers

Posix I/O

Split Files

MPI I/O

Custom

Storage

HDF5 File Format

File

Split Files

File on Parallel Filesystem

? Other

Useful Tools For New Users

h5dump:

Tool to “dump” or display contents of HDF5 files

h5cc, h5c++, h5fc:

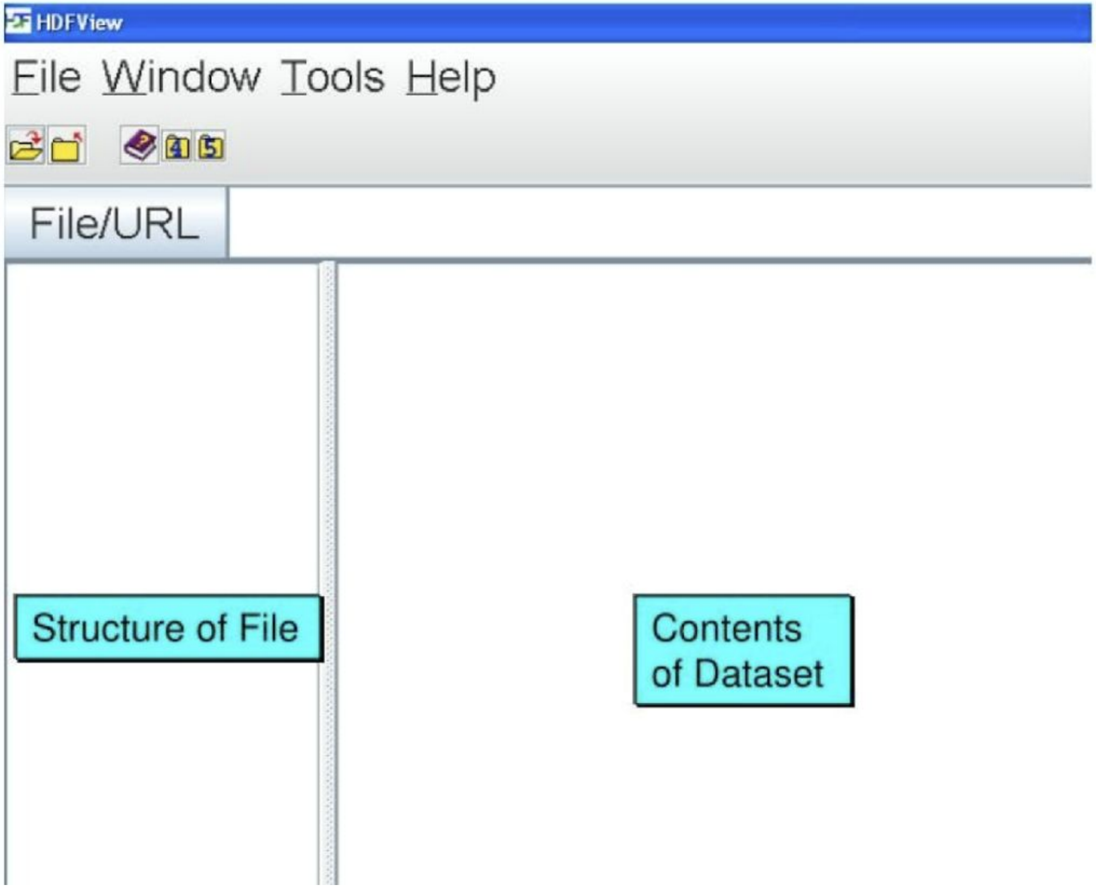
Scripts to compile applications

HDFView:

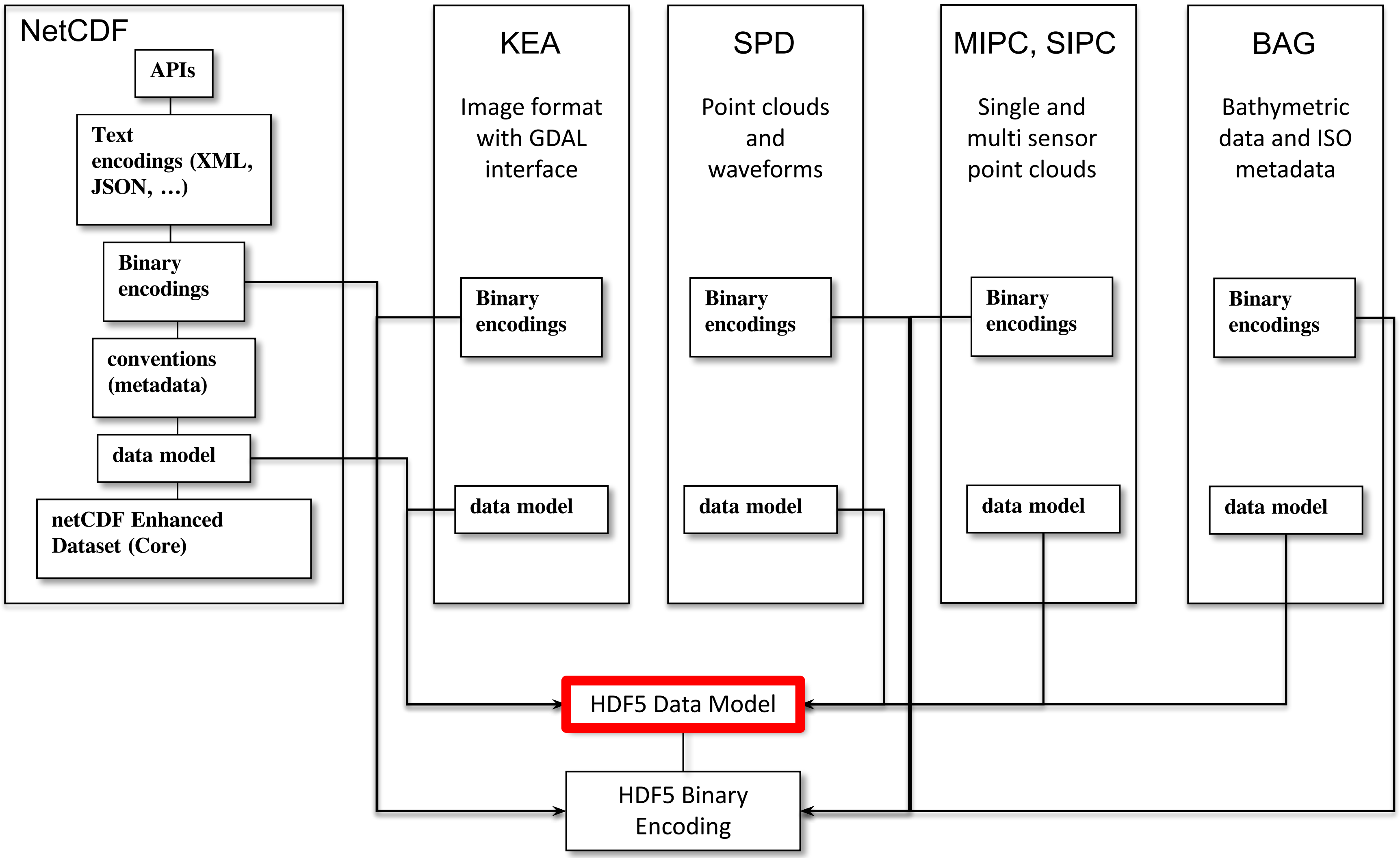
Java browser to view HDF4 and HDF5 files

http://www.hdfgroup.org/hdf-java-html/hdfview/

HDFView


HDFView
File Window Tools Help
File/URL
Structure of File
Contents of Dataset

Where HDF5 Fits in OGC (Present)



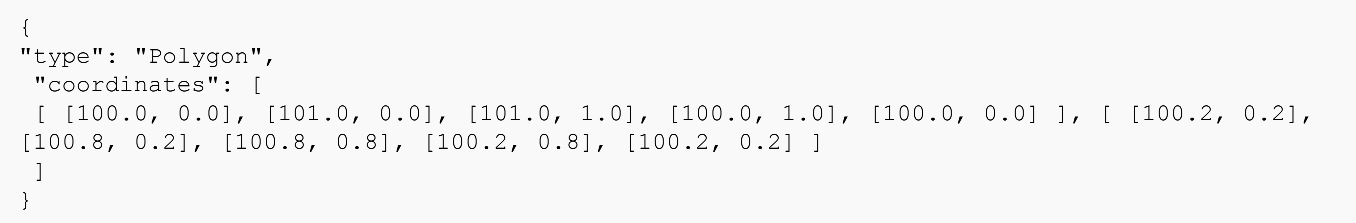
HDF4 -> HDF5

- How is it different
  - HDF5 data model is "simpler" with less but consistent object model
  - HDF4 and HDF5 Data Models and APIs are considerably different
    - HDF5 users do not worry about file formats or SD, DFR8,DF24, GR, Vdata, Vgroup …
  - HDF5 high-level APIs provide simpler interfaces for common tasks.
  - HDF4 objects can all be conceptually mapped to an HDF5 Group or Dataset
- What features does HDF5 have that HDF4 does not?
  - **An HDF5 file has a true hierarchical file structure and a naming scheme for all the objects in the file (path names).**
  - HDF5 has "Dataspace" objects, which are not in HDF4.
  - HDF5 has "Datatype" objects, which are different than HDF4 datatypes.
  - HDF5 supports multiple storage models and it supports MPI-IO.
  - ***But no***
    - read/write netCDF with HDF5
    - Jpeg or other compression

QUIZ

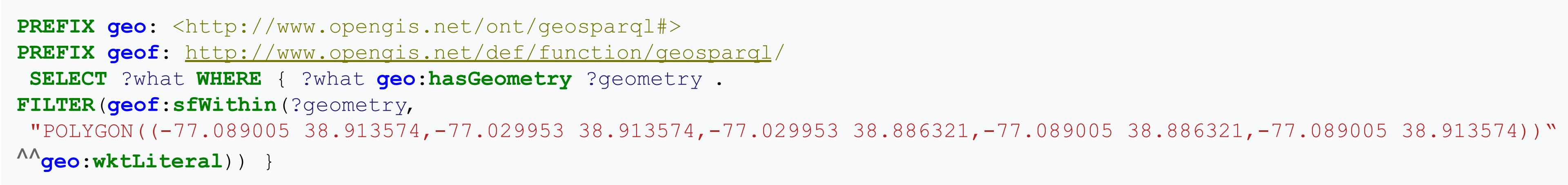
- Why do we use
  - GeoJSON
  - GeoSPARQL
  - HDF5
  - GML

- What are these



```
{
"type": "Polygon",
 "coordinates": [
 [ [100.0, 0.0], [101.0, 0.0], [101.0, 1.0], [100.0, 1.0], [100.0, 0.0] ], [ [100.2, 0.2],
[100.8, 0.2], [100.8, 0.8], [100.2, 0.8], [100.2, 0.2] ]
 ]
}
```

```
<gml:Polygon> <gml:outerBoundaryIs> <gml:LinearRing>
<gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
</gml:LinearRing> </gml:outerBoundaryIs> </gml:Polygon>
```



```
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: http://www.opengis.net/def/function/geosparql/
 SELECT ?what WHERE { ?what geo:hasGeometry ?geometry .
FILTER(geof:sfWithin(?geometry,
 "POLYGON((-77.089005 38.913574,-77.029953 38.913574,-77.029953 38.886321,-77.089005 38.886321,-77.089005 38.913574))“
^^geo:wktLiteral)) }
```

Lunch

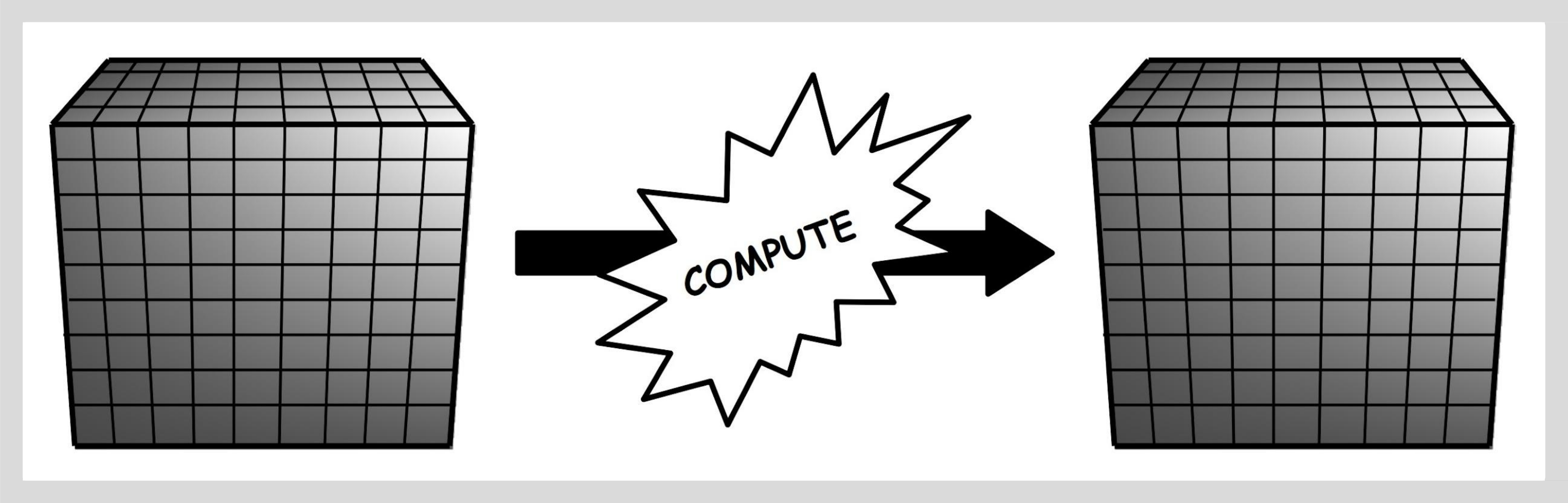
Make sure you have

- Downloaded HDF5 viewer
- Looked up the tutorial at https://portal.hdfgroup.org/display/HDF5/HDF5+Examples
- Downloaded some of the data from
  - http://hdfeos.org/zoo/index_openGESDISC_Examples.php
  - Intro to Working with Hyperspectral Remote Sensing Data in HDF5 Format in R | NSF NEON | Open Data to Understand our Ecosystems (neonscience.org)
- Visualized the data in QGIS

Zarr

STAC
SpatioTemporal Asset Catalog
Zarr
COPC.io
Cloud Optimized Point Cloud
COG
CLOUD OPTIMIZED GEOTIFF

Problem statement



There is some computation we want to perform.

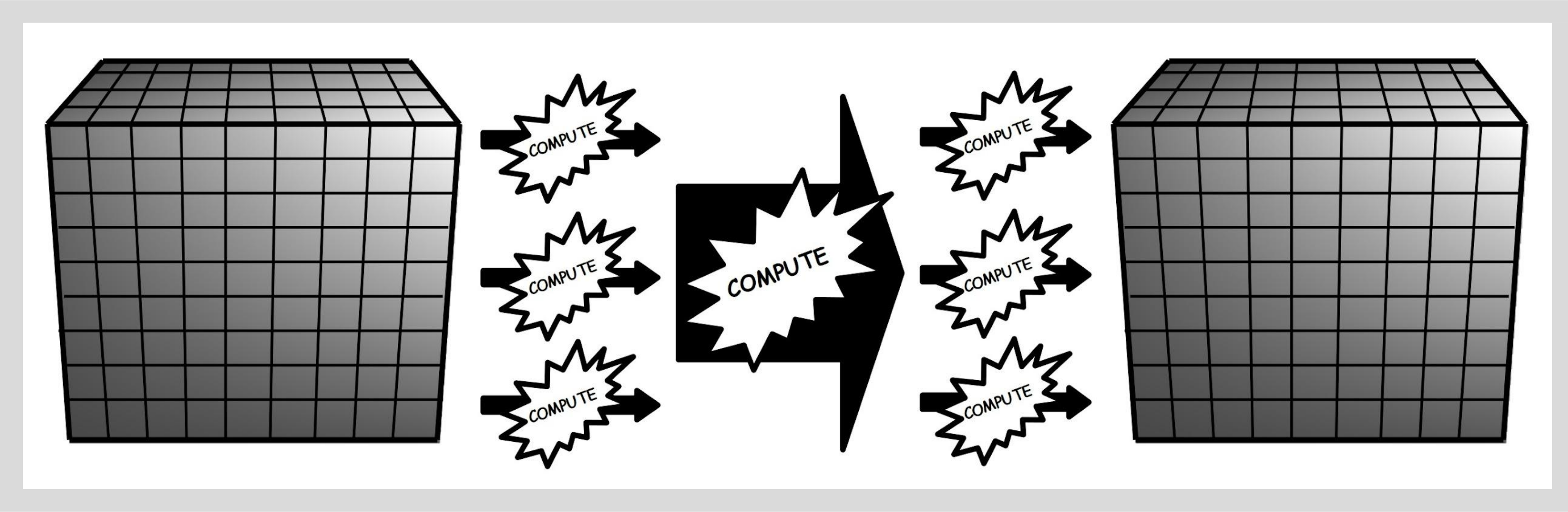
Inputs and outputs are multidimensional arrays (a.k.a. tensors).

5 key features...

(1) Larger than memory

Input and/or output tensors are too big to fit comfortably in main memory.

(2) Computation can be parallelised



At least some part of the computation can be parallelised by processing data in chunks.

(3) I/O is the bottleneck

Computational complexity is moderate → significant amount of time is spent in reading and/or writing data.

N.B., bottleneck may be due to (a) limited I/O bandwidth, (b) I/O is not parallel.

(4) Data are compressible

Compression is a very active area of innovation. Modern compressors achieve good compression ratios with very high speed.
Compression can increase effective I/O bandwidth, sometimes dramatically.

(5) Speed matters

Rich datasets → exploratory science → interactive analysis → many rounds of summarise, visualise, hypothesise, model, test, repeat.

E.g., genome sequencing.

Now feasible to sequence genomes from 100,000s of individuals and compare them.

Each genome is a complete molecular blueprint for an organism → can investigate many different molecular pathways and processes. Each genome is a history book handed down through the ages, with

each generation making its mark → can look back in time and infer major demographic and evolutionary events in the history of populations and species.

Zarr data in Web Maps

https://maps.demo.carbonplan.org/

Short Tutorial on Zarr

HDF5

- over 20 years old
- excellent cross language support
- lots of existing software
- written in (very clean) C
- can be made thread safe, not thread optimal
- extensible in C

Zarr

- first release in 2015, 1.0 on May 17, 2016
- written in Python, Python oriented
- has specification which could be reimplemented
- multithreading support
- extensible in Python

Today

0930-1000	Recap	Participants			
1000-1130	Graph DB walkthough	Sumit Sen	Slides from Sumit	GeoSPARQL	
1130-1145	Tea				
1145-1300	HDF5	Sumit Sen	Slides from Sumit	GitHub - HDFGroup/hdf5: Official HDF5® Library Repository	
1300-1400	Lunch	-	-		
1400-1530	Zarr	Sumit Sen	Slides from Sumit		
1530-1545	Tea				
1545-1700	CDB	Sumit Sen	Slides from Sumit		

CDB

- The OGC CDB (Common Database) standard defines a standardized model and structure for a single, “versionable”, virtual representation of the earth.

Motivation: Mission Rehearsal

Carl Philipp Gottfried von Clausewitz
18th century Prussian soldier and German military theorist

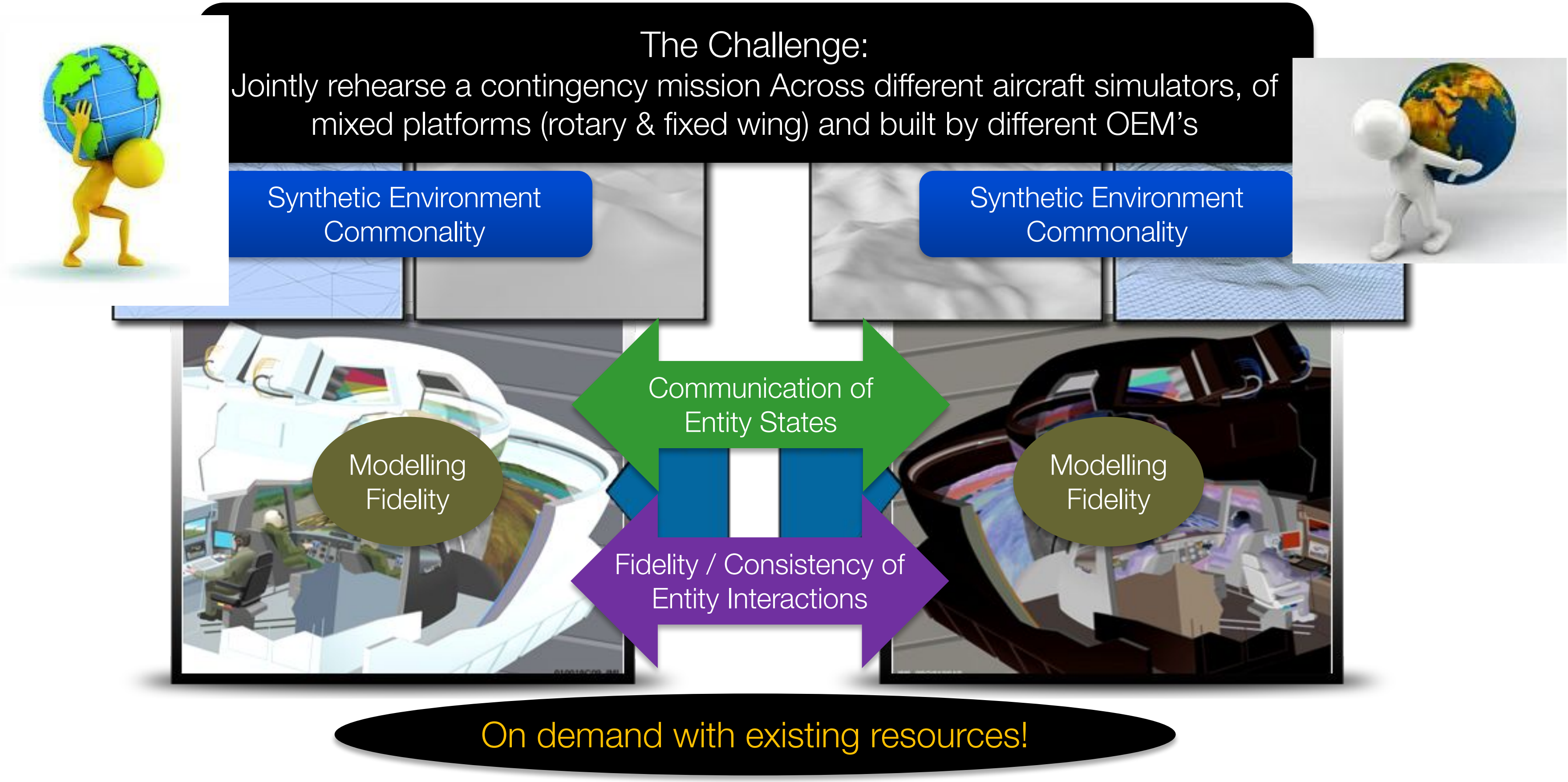
> *"…it is of immense importance that the soldier should not have to encounter in War those things which, when seen for the first time, set him in astonishment and perplexity;*
>
> *…if he has only met with them one single time before even by that he is half acquainted with them…"*

Situational Awareness - Key to Effective Mission Rehearsal

Effective Interoperability

Agility of Re-Configurability

Motivation: Mission Rehearsal



The Challenge:
Jointly rehearse a contingency mission Across different aircraft simulators, of mixed platforms (rotary & fixed wing) and built by different OEM's

Synthetic Environment Commonality

Synthetic Environment Commonality

Modelling Fidelity

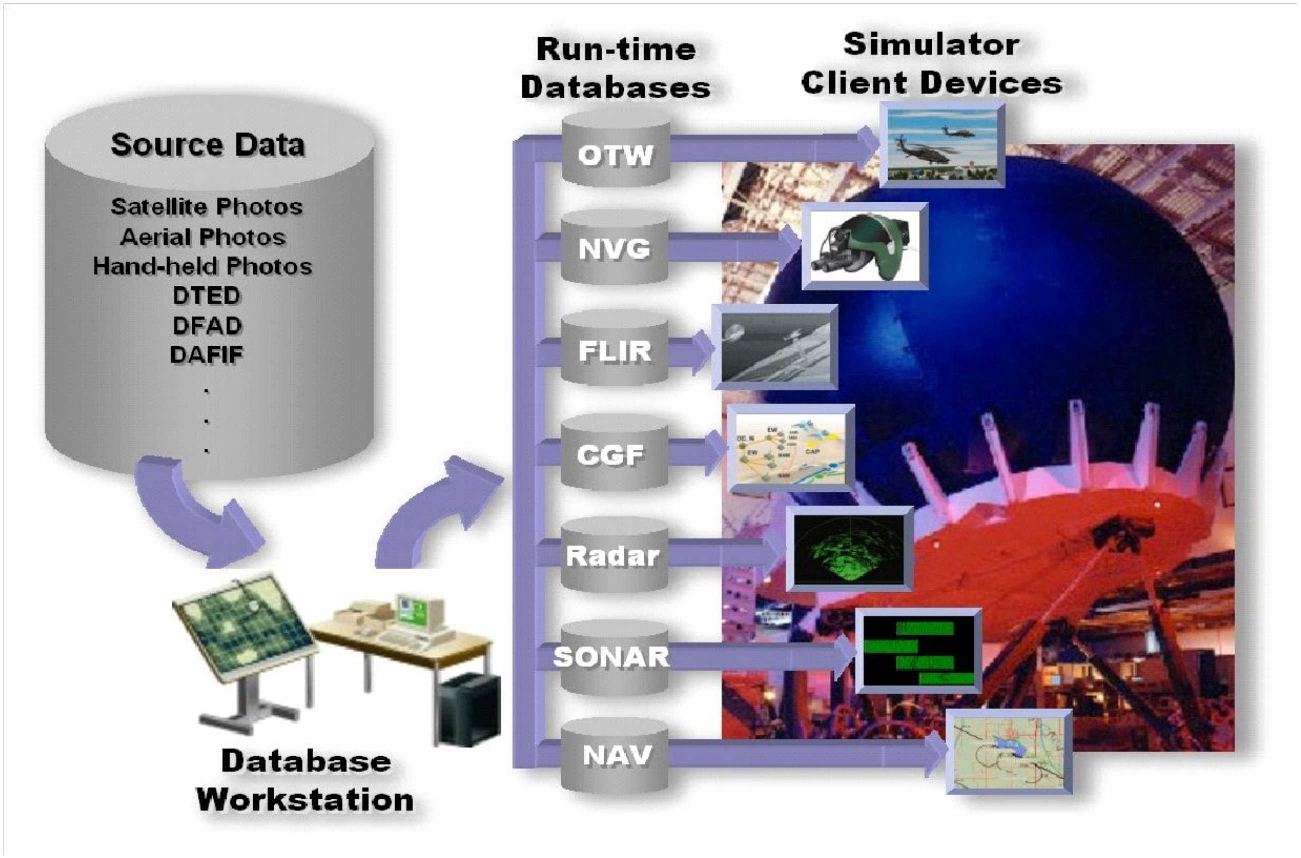
Communication of Entity States

Fidelity / Consistency of Entity Interactions

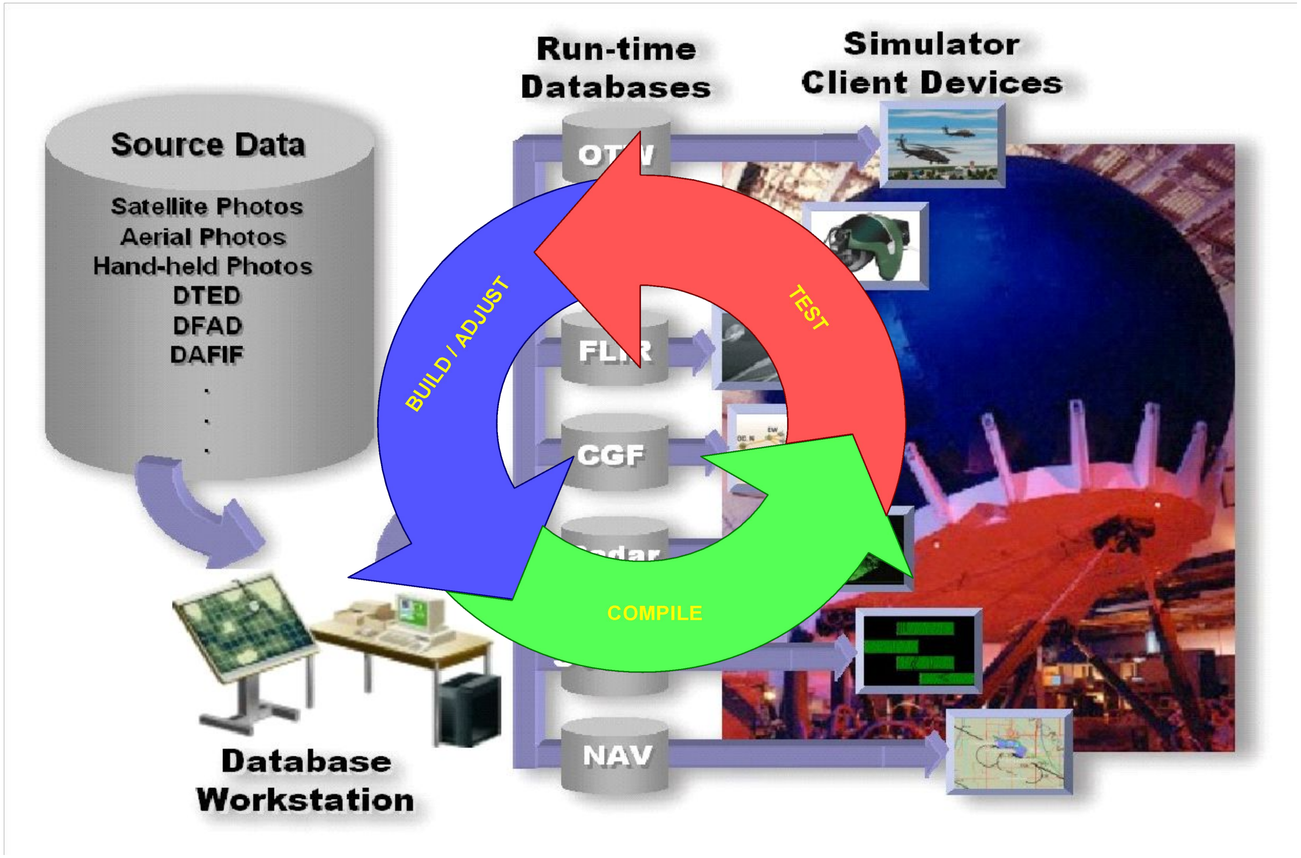
Modelling Fidelity

On demand with existing resources!

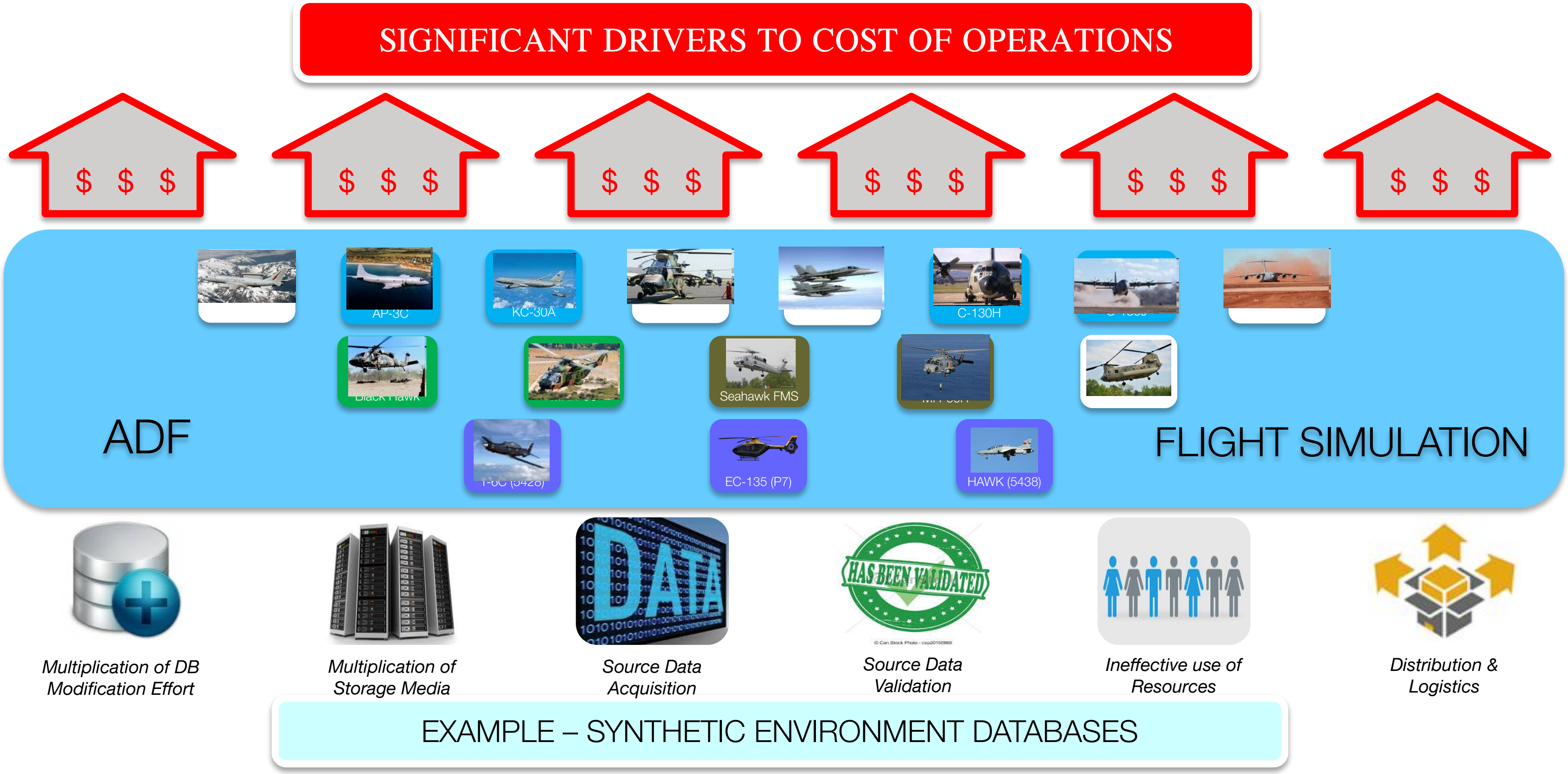
Flight Simulation – Synthetic Environments



Flight Simulation – Synthetic Environments



ADF Flight Simulation – Current Context



SIGNIFICANT DRIVERS TO COST OF OPERATIONS

ADF

FLIGHT SIMULATION

Multiplication of DB Modification Effort

Multiplication of Storage Media

Source Data Acquisition

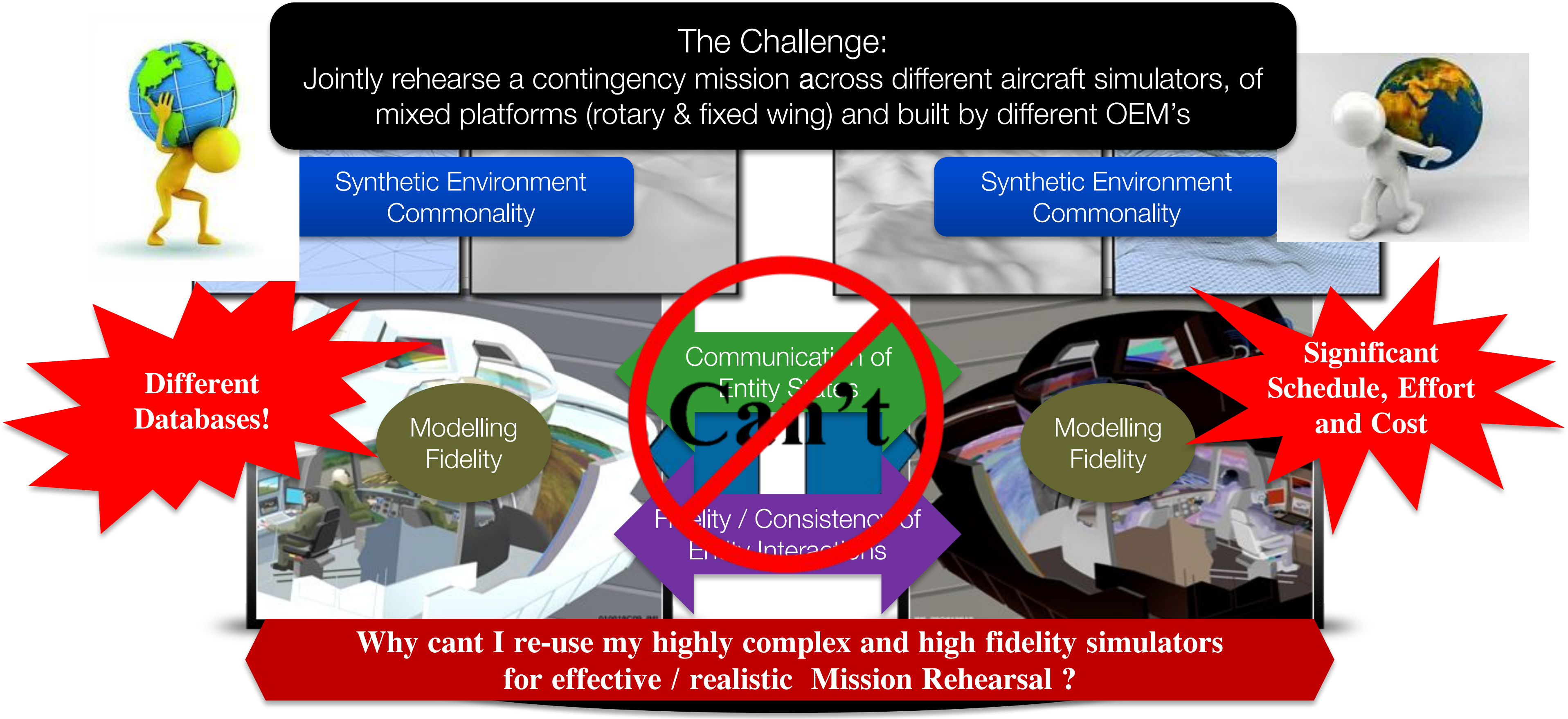
Source Data Validation

Ineffective use of Resources

Distribution & Logistics

EXAMPLE – SYNTHETIC ENVIRONMENT DATABASES

Motivation: Mission Rehearsal



The Challenge:
Jointly rehearse a contingency mission across different aircraft simulators, of mixed platforms (rotary & fixed wing) and built by different OEM’s

Synthetic Environment Commonality

Synthetic Environment Commonality

Different Databases!

Modelling Fidelity

Communication of Entity States

Can’t

Fidelity / Consistency of Entity Interactions

Modelling Fidelity

Significant Schedule, Effort and Cost

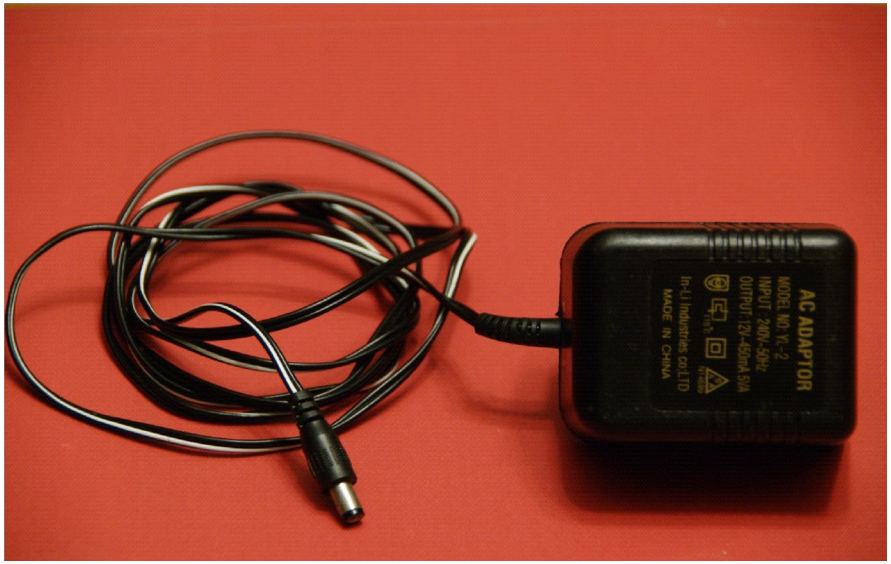
Why cant I re-use my highly complex and high fidelity simulators for effective / realistic Mission Rehearsal ?

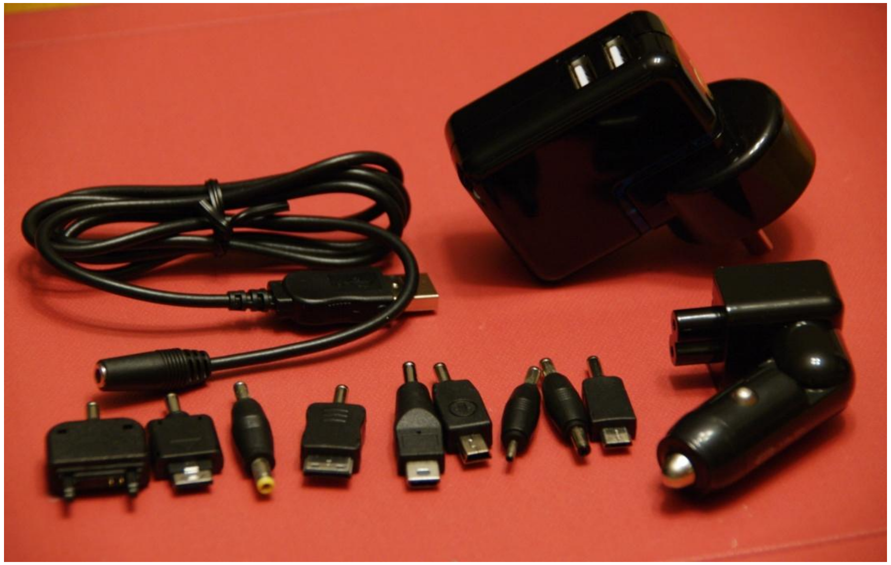
FROM THE ACADEMY AWARD-WINNING
WRITER AND DIRECTOR OF THE HURT LOCKER
ZERO DARK THIRTY
THE GREATEST MANHUNT IN HISTORY
COLUMBIA PICTURES PRESENTS A MARK BOAL PRODUCTION A FIRST LIGHT PRODUCTION AN ANNAPURNA PICTURES PRODUCTION A KATHRYN BIGELOW FILM
"ZERO DARK THIRTY" JESSICA CHASTAIN JASON CLARKE JOEL EDGERTON CASTING BY MARK BENNETT CSA RICHARD HICKS CSA GAIL STEVENS MUSIC BY ALEXANDRE DESPLAT
SOUND DESIGN PAUL N.J. OTTOSSON COSTUME DESIGNER GEORGE L. LITTLE EDITED BY DYLAN TICHENOR, A.C.E. WILLIAM GOLDENBERG, A.C.E. PRODUCTION DESIGNER JEREMY HINDLE DIRECTOR OF PHOTOGRAPHY GREIG FRASER ACS
EXECUTIVE PRODUCERS COLIN WILSON TED SCHIPPER GREG SHAPIRO PRODUCED BY MARK BOAL KATHRYN BIGELOW MEGAN ELLISON WRITTEN BY MARK BOAL
R RESTRICTED
DIRECTED BY KATHRYN BIGELOW
SONY make.believe
COLUMBIA PICTURES
JANUARY
ZeroDarkThirty-Movie.com

Competitive Award from the US Army Program Office
For the
United States Special Operations Command (USSOCOM)

FROM THE ACADEMY AWARD® WINNING
WRITER AND DIRECTOR OF THE HURT LOCKER
ZERO DARK THIRTY
THE GREATEST MANHUNT IN HISTORY
SONY
COLUMBIA PICTURES
JANUARY
ZeroDarkThirty-Movie.com

Need for Architectural Innovation

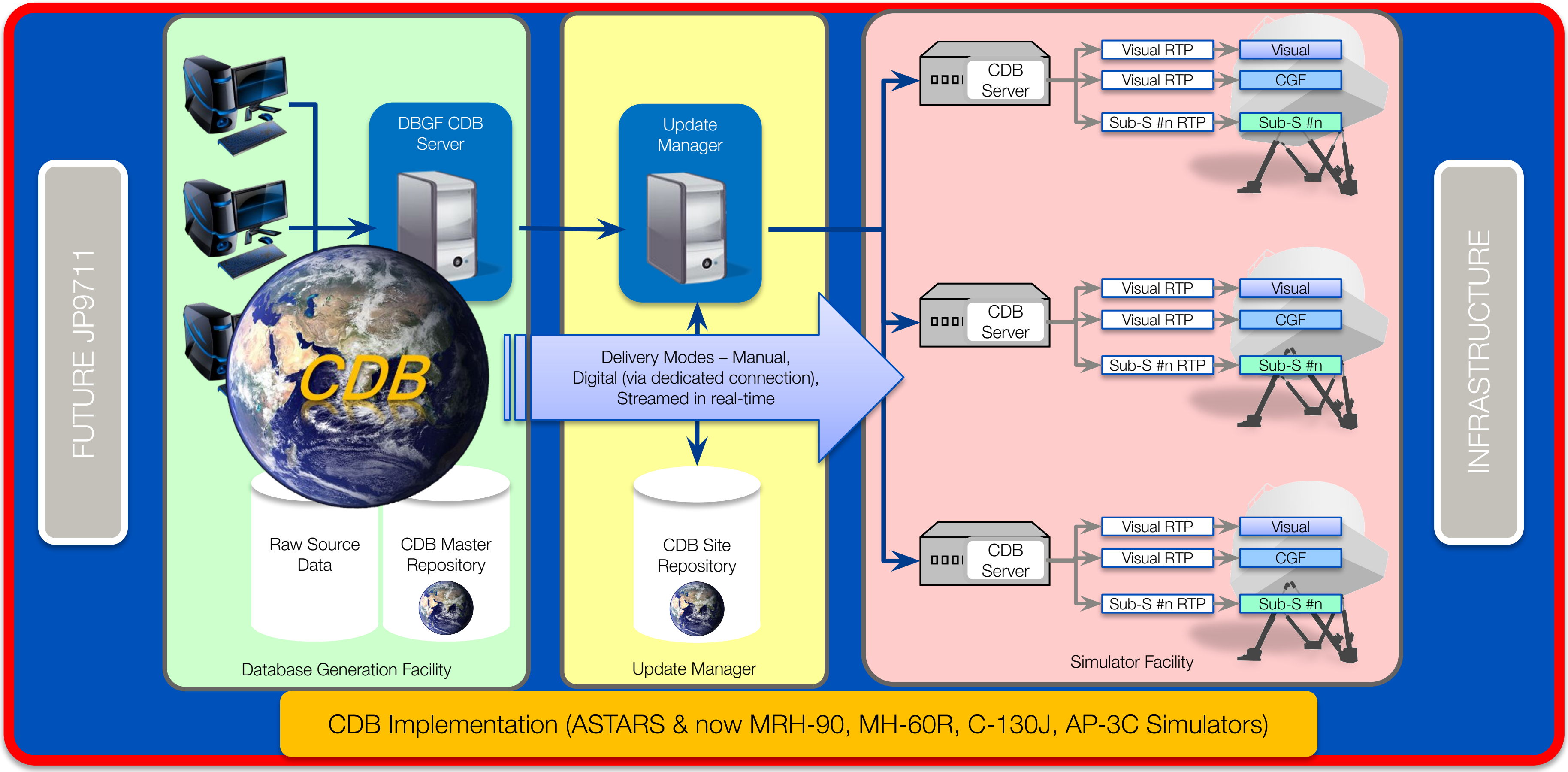




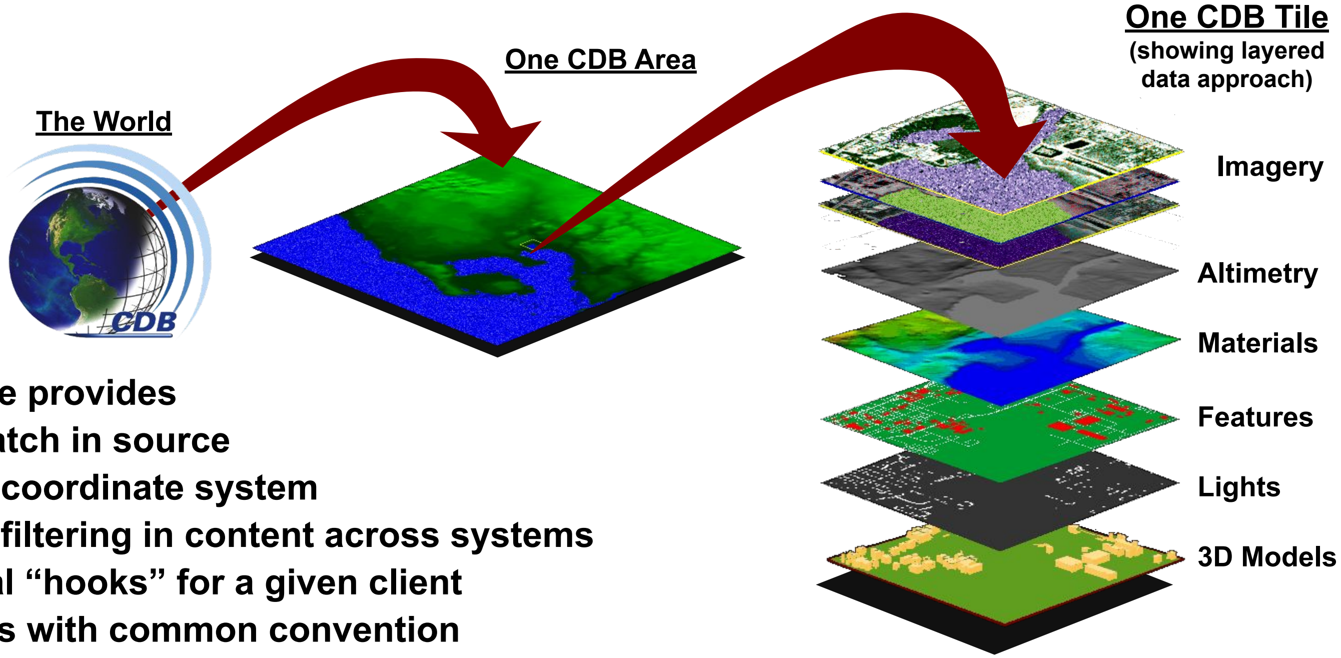
Compare the products above in the contexts of

Interoperability? Agility of Re-configurability? Supportability ? Open Architecture ? IP Protection ?

Need for Architectural Innovation – Separating Content from Use



Need for Architectural Innovation - An Open CDB Format

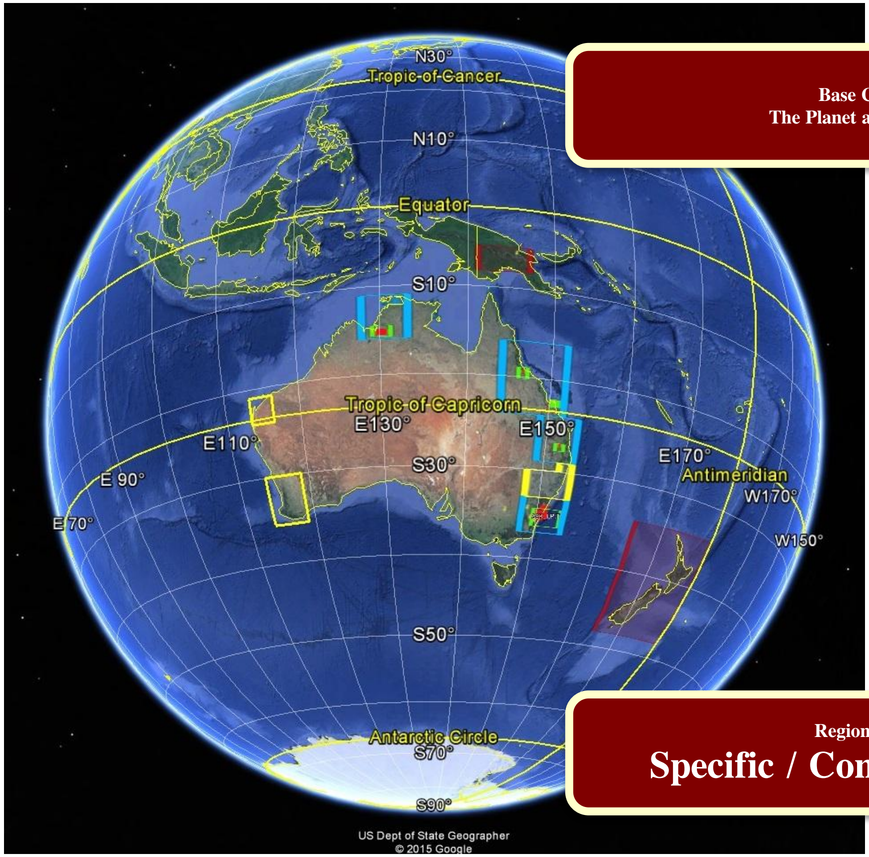


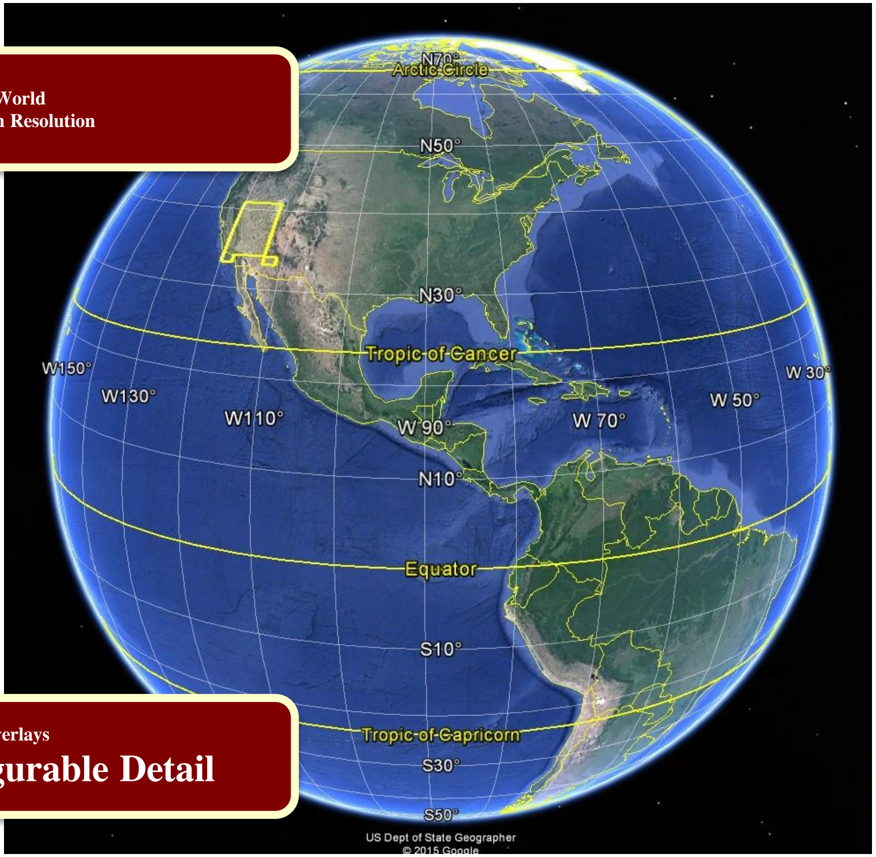
CDB structure provides

- No mismatch in source
- Common coordinate system
- Common filtering in content across systems
- No special “hooks” for a given client
- 3D models with common convention

CDB provides both correlated content across systems and inherent interoperability support & agility of re-configurability

Summary : CDB SEDB World with Regional Overlays





Base CDB World
The Planet at 15m Resolution

Regional Overlays

Specific / Configurable Detail

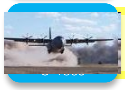
CDB SEDB's Compatible Simulators and Regional Overlays: Australia, PNG, NZ

Beech 350 - Sale

MRH-90 – Oakey & Townsville

HAWK – Williamtown & Pearce

AP-3C - Edinburgh



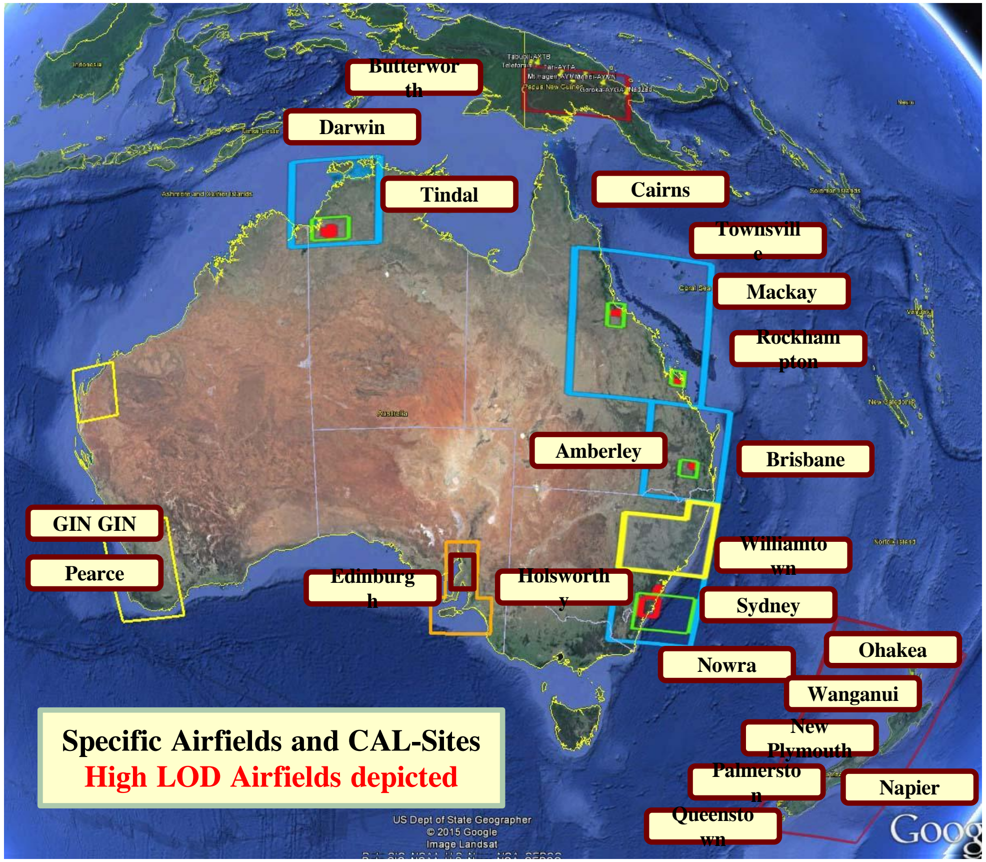
C-130J - Richmond

MH-60R - Nowra

NZDF T-6

NZDF Super Seasprite

KEY	Low LOD TA DTED1, 15m-2.5m (Imagery), No 3D	Medium LOD TA DTED1, 2.5m-1.0m (Imagery), 3D Trees	High LOD TA DTED1, 1.0m-0.5m (Imagery), 3D Features

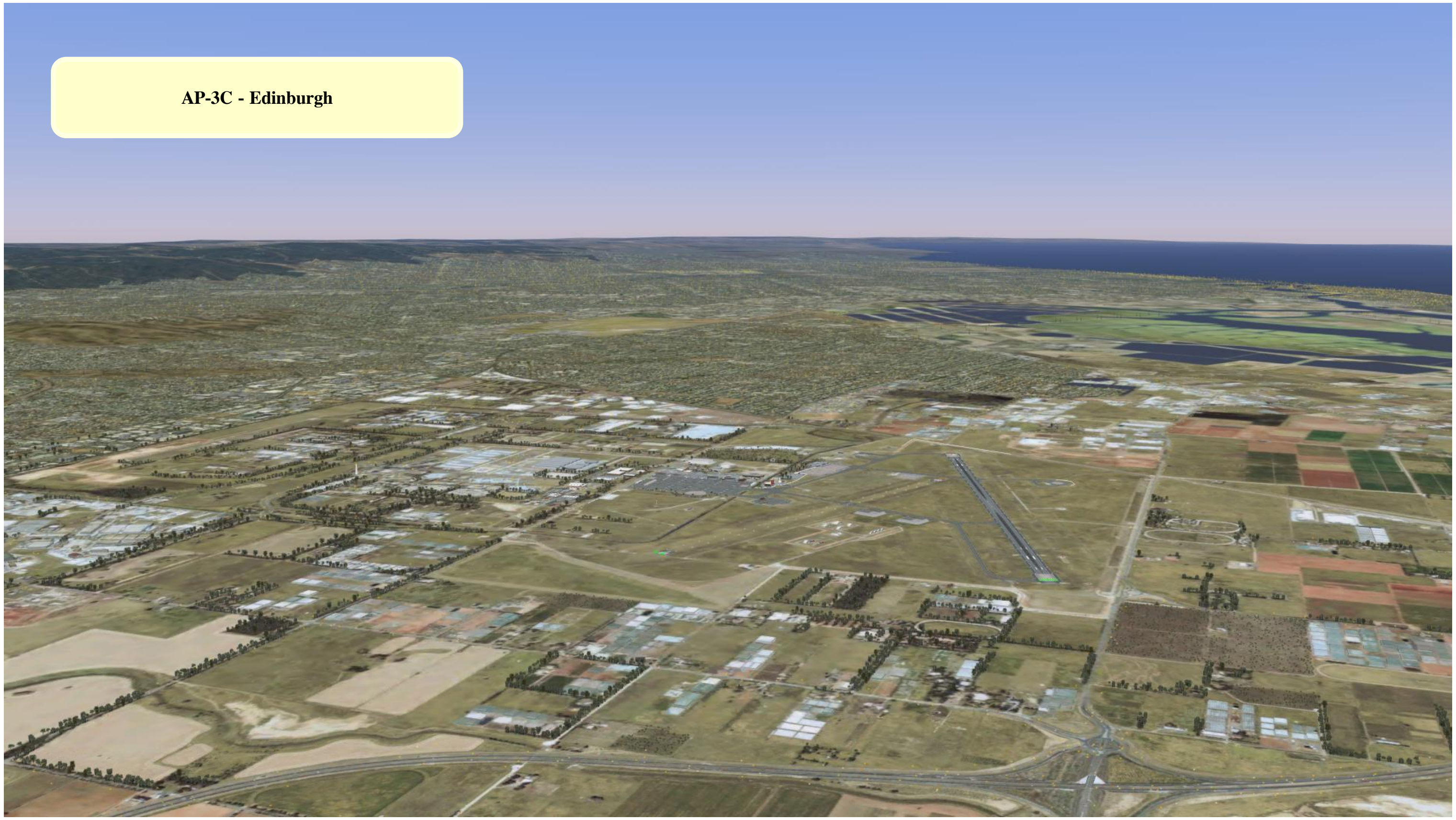


Specific Airfields and CAL-Sites

High LOD Airfields depicted

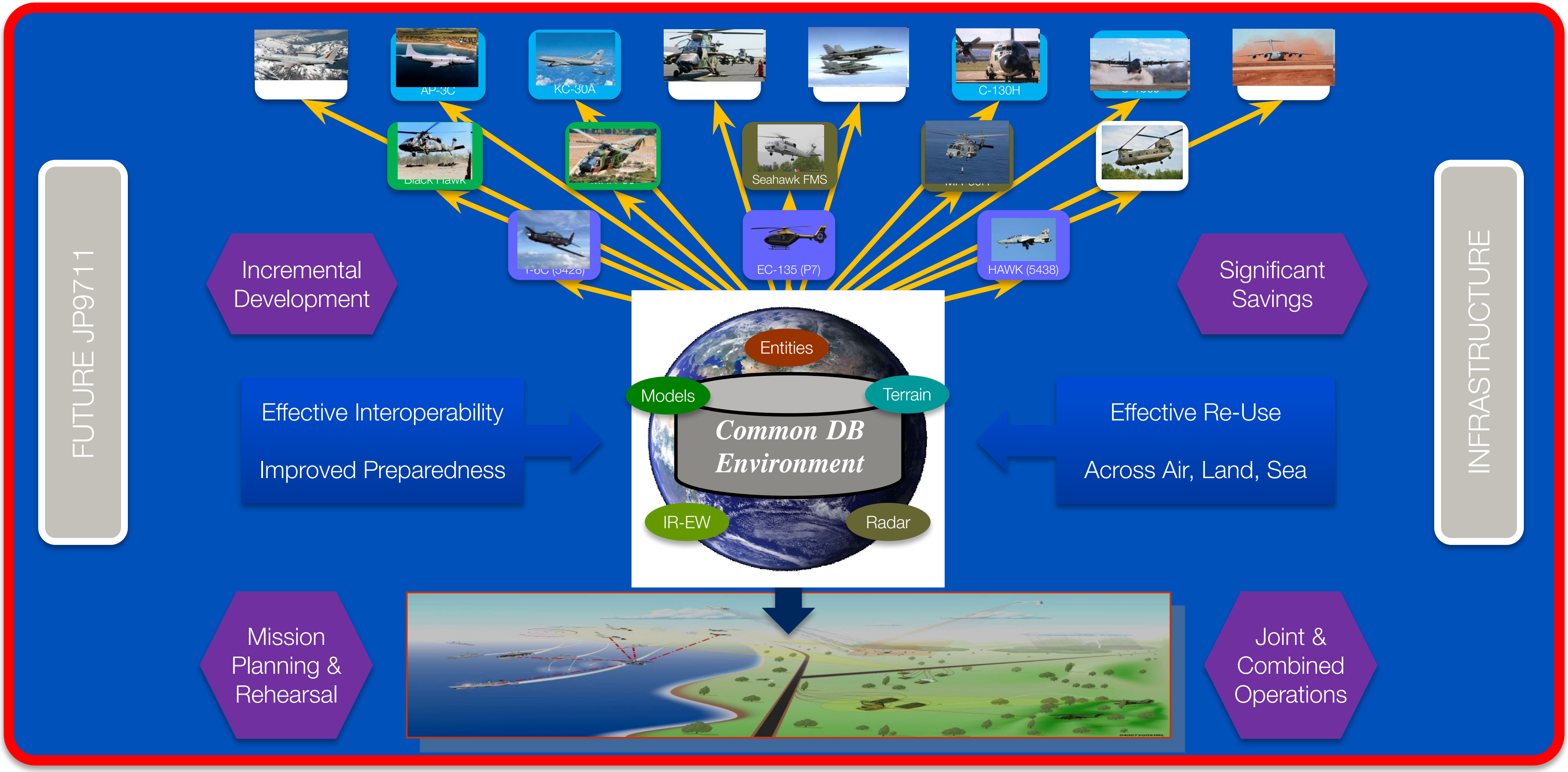
US Dept of State Geographer
© 2015 Google
Image Landsat

CDB SEDB's Compatible Simulators and Regional Overlays: Edinburgh



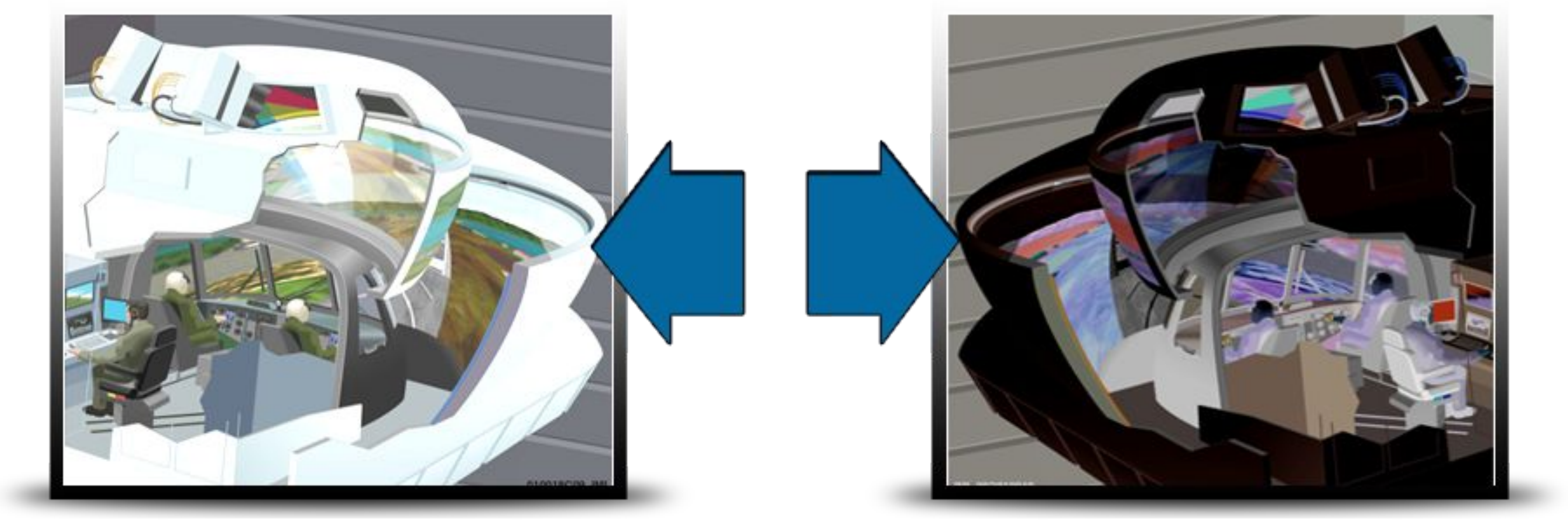
- CAE's Vision for the ADF

ADF Vision SEDB Enabling Effective



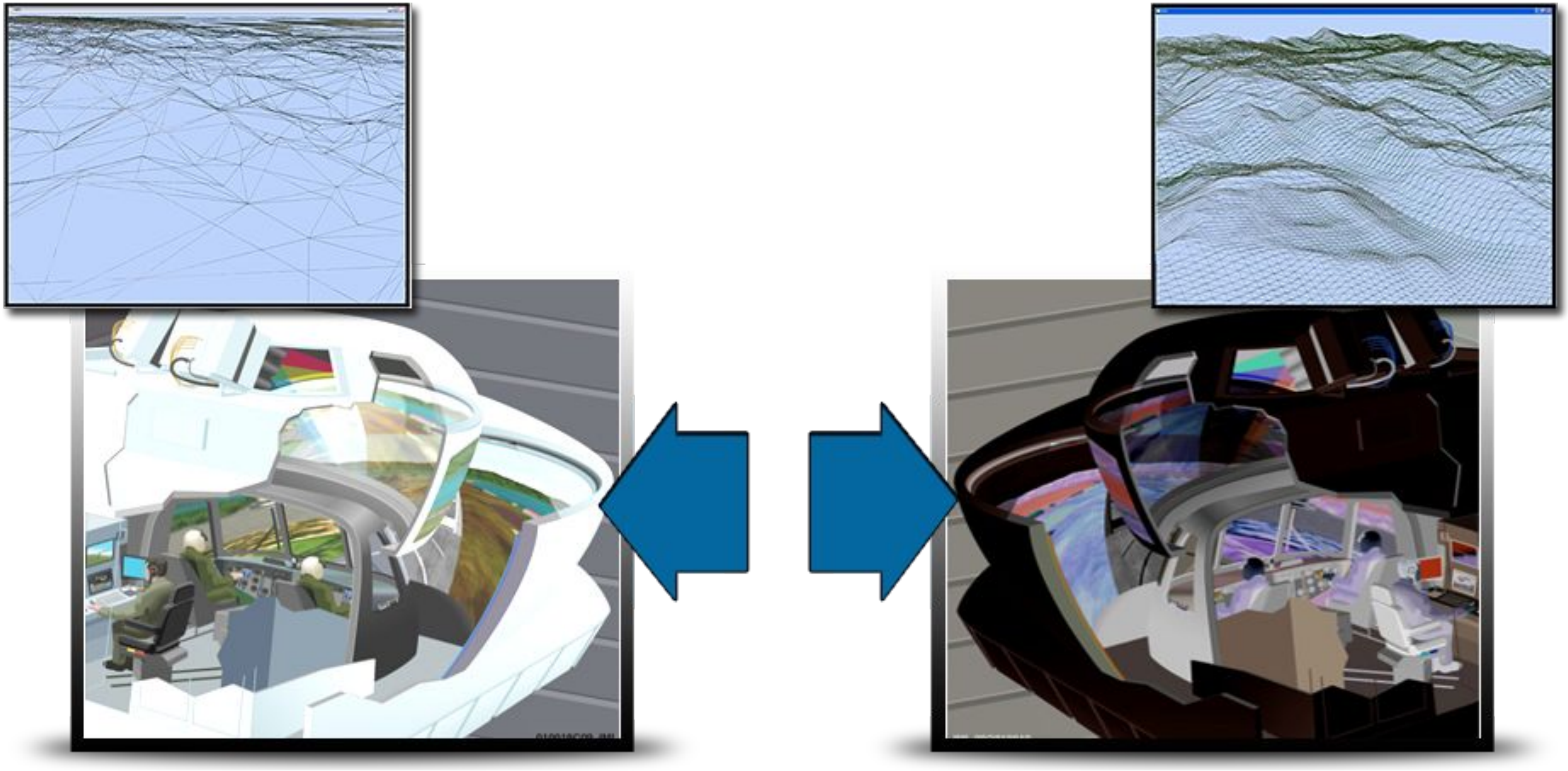
Simulation Interoperability - Connectivity

- For at least 20 years or so, there has been general agreement about how to ‘connect’ simulators:
  - Distributed Interactive Simulation (DIS)
  - High Level Architecture (HLA)
    - These standards allow simulation systems to communicate data about the positions and states of simulated entities



Simulation Interoperability – local representation of the Environment

There has never been any standardization, in the actual run-time simulators, of the data sets that describe the physical world and environment



CDB is an existing de-facto specification for high performance simulation data bases

COMMON DATABASE SPECIFICATION

Volume 1

Version 3.2

19 March 2014

ION

ASE

TION

Volume 2

Version 3.2

19 March 2014

Multiple, Independent implementations starting In 2007

CDB-based simulators are In: USA, Canada, UK, Germany, Turkey, Israel, Singapore, Australia, Brunei and other countries

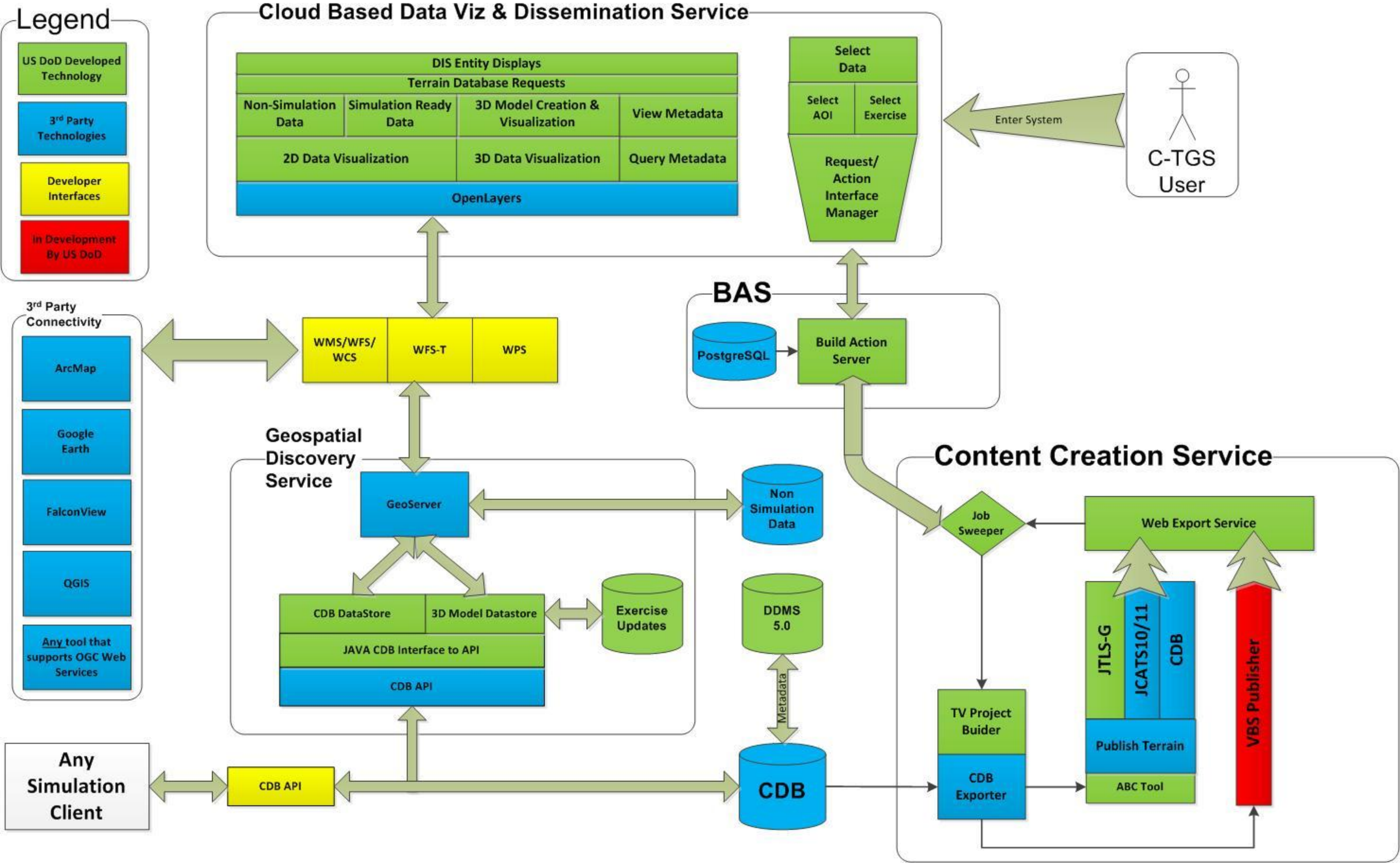
A video of CDB simulation ‘clients’ rendering a CDB:
http://youtu.be/zxzPAcP6Ygo

CDB in the Microsoft Azure Cloud
http://geoserver1demo.cloudapp.net/

So, What is a CDB Conformant Data Store?

- A CDB data store contains the features and modeled representation of the synthetic environment. A CDB data store can contain:
- terrain altimetry,
- planimetry,
- raster imagery,
- attribution,
- 3D features with their modeled geometry,
- texture and attribution.

This is implementation architecture for C-TGS. Implementation software represents a mix of commercial/proprietary and open source. At the heart of the C-TGS are two data stores: A CDB structured data store and a 3D model data store



Legend

- 🟩 US DoD Developed Technology
- 🟦 3rd Party Technologies
- 🟨 Developer Interfaces
- 🟥 In Development By US DoD

Cloud Based Data Viz & Dissemination Service

DIS Entity Displays
Terrain Database Requests
Non-Simulation Data
Simulation Ready Data
3D Model Creation & Visualization
View Metadata
2D Data Visualization
3D Data Visualization
Query Metadata
OpenLayers

Select Data
Select AOI
Select Exercise
Request/ Action Interface Manager

Enter System

C-TGS User

3rd Party Connectivity

ArcMap
Google Earth
FalconView
QGIS
Any tool that supports OGC Web Services

WMS/WFS/ WCS
WFS-T
WPS

BAS

PostgreSQL
Build Action Server

Geospatial Discovery Service

GeoServer
CDB DataStore
3D Model Datastore
JAVA CDB Interface to API
CDB API
Exercise Updates

Non Simulation Data
DDMS 5.0
Metadata
CDB

Any Simulation Client
CDB API

Content Creation Service

Job Sweeper
Web Export Service
JTLS-G
JCATS10/11
CDB
Publish Terrain
ABC Tool
VBS Publisher
TV Project Buider
CDB Exporter

Some links

- https://gaj-geospatial.com/CDB_Plugin
- https://www.youtube.com/watch?v=euHi47OI1kI
- GNOSIS Map Server @ maps.gnosis.earth (ecere.com)