Heterogeneous Data Integration: Challenges, Approaches & Benefits

Derek Hohls, Graeme McFerren & Bolelang Sibolla
ISRSE — May 9, 2017
What is Required?

- **Access** data from multiple sources
- **Integrate** data via consistent, common formats
- **Expose** data via common, well-known interfaces
- **Enable** support for modelling, analysis & visualisation

Timeously observe the earth "in motion"
Challenges for Observational Data

1. Multiple Sources
2. Multiple Formats
3. Differing Time Scales
4. Differing Spatial Scales
5. Multiple Characteristics
6. Asynchronous Data Feeds
Approaches to Data Integration

- Standards-based formats
  - Open Geospatial Consortium: Sensor Observation Service (SOS)
  - Common Data Model

- Data Pipelines
  - Per use-case scripts/modules for data processing

- Tooling
  - Sensor data: 52N SOS Implementation
  - Streaming data: swordfish Python library
swordfish: data streaming framework (Python)

- Supports key functions for processing "data-on-the-move"
  - Rapid recognition of, and access to, spatial components of data
- Handles all data via internal Common Data Model based on
  - GeoJSON
  - Unidata Common Data Model
  - OGC O&M
- Establishes live streams - including ephemeral, episodic & continuous types
swordfish: topology

source → filter → map → datastore
source → 

map → join → reduce

socket → alert

stream → processor → source / sink

componentry
Background to Case Studies

- **Common problems**
  - Data comes from different types of sources
  - Data usually available in non-standard formats
  - Data "consumers" differ in requirements

- **Common approach**
  - Establish purpose/use of data
  - Understand data source (type; format; frequency)
  - Attach to the data source(s) & transform to a common data model
  - Make data available in standard formats for other off-the-shelf tools
Case Study #1: Beach Water Quality

Data Source: static files (spreadsheets) with laboratory data

Desired Usage: display changes in water quality per beach

Approach:

- Retrieve and iterate over files from server
- Data stream $\Rightarrow$ SOS Ingestion Module
- Sink data to a SOS-compliant data store
- Retrieve data from SOS via standard interface
  - Map beach locations (OpenLayers)
  - Graph water quality data (D3 Visualisation library)
This website provides information about the current and historic conditions of the Blue Flag beaches in South Africa, based on measurable environmental criteria. Please click a location on the map to view available measurements, then click on the property of choice to view the time series.

A Blue Flag is an international award given to beaches, boats, and marinas that meet EXCELLENCE in the areas of safety, amenities, cleanliness, and environmental standards. The strict criteria of the programme are set by the International co-ordinators of the Blue Flag campaign in Europe, the Foundation for Environmental Education (FEE).

- **eColi**
  - Refers to faecal coliforms which test indole-positive at 44.5°C, and generally consists only of E. coli which is almost definitely of faecal origin. Used to evaluate the possible faecal origin of total and faecal coliforms, usually when these are isolated from drinking water.
  - **entero**
  - **float**
  - **oil**
  - **pH**
  - **salinity**
Case Study #2: Electricity Metering  

(Continuous Data)

- **Data Source:** off-campus data repository
- **Desired Usage:** view near-realtime consumption & trends
- **Approach:**
  - Access live data repository
  - Create *swordfish* module for SOAP data access
  - Map data to Common Data Model
  - Structure data as JSON for transport via web-sockets to downstream clients
  - Enable long-term storage in a SOS-compliant data store
Case Study #3: Lightning Detection  (Ephemeral Data)

- **Data Source:** cloud database & in-situ sensors
- **Desired Usage:** visualise lightning strikes in real-time
- **Approach:**
  - Lightning database change triggers event
  - Event transport to message broker
  - Stream & map data to Common Data Model
  - Structure data as JSON for dissemination via web-sockets to downstream clients for visualisation
What is Possible?

● No "magic" one-size-fits-all solution

● Limit "pain points" - for example:
  ○ Internal processing via common data model

● System development via chained components
  ○ Specialised: per-project
  ○ Common/Reusable: standards-based

● Downstream users access well-known formats/interface; e.g.
  ○ JSON
  ○ SOS (XML)
Way Forward

- Ongoing development of *swordfish* capabilities
- Apply tools and approaches further into our oceans environment systems e.g.
  - Automatic Identification System (AIS) data stream
  - In-situ buoys and sensors
  - Ad-hoc site monitoring
  - Ocean state forecast models
  - Social media streams
Acknowledgements

- **Funding**
  - CSIR
  - DBSA
  - Department of Science & Technology
  - Department of Environment Affairs / *Operation Phakisa*

- **Data Sources**
References

- OGC Sensor Observation Service (http://www.opengeospatial.org/standards/sos)
- GeoJSON (http://geojson.org/)
Thank you!

Any questions?
PostScript #1: Common Data Model example

```json
{
  "type": "CF_SimpleObservation",
  "phenomenonTime": "2016-09-14T11:58:58.649",
  "result": {
    "variables": {
      "pulse_value": {"units": "litres", "dimensions": ["int"]},
      "time": {"units": "isoTime", "dimensions": ["time"]},
      "data": {"pulse_value": [500, 500],
                "time": ["2016-09-14T11:58:01.305", "2016-09-14T11:47:06.808"]},
      "dimensions": {"time": 2})),
    "featureOfInterest": {
      "geometry": {"type": "Point", "coordinates": [25.753, 28.28]},
      "type": "Feature",
      "properties": {"id": "Reservoir"}},
    "observedProperty": {"type": "TimeSeries"},
    "procedure": {"type": "sensor",
                  "id": "45030171",
                  "description": "Sensus HRI Mei meter 14787486"}
  }
}
```
Postscript #2: Data-on-the-move

- **Transport**: keep the data moving!
- **Extract**: deserialise // decode
- **Translate**: clean // re-format // re-model
- **Load/unload**: sink to database // pass to process // serialise
- **Analyse**: anomaly detection // complex event processing
- **Output**: generate new data
PostScript #3: Outline

- What is Required?
- Challenges for Sensed & Sensor Data
- Approaches to Data Integration
- Handling "data on the move"
- Case Studies
- What is Possible
- Way Forward