Marine National Facility

RVIDAS

Novel Additions to OpenRVDAS on the RV Investigator

2023

Ella Pietraroia

Operated by CSIRO, Australia's National Science Agency, on behalf of the nation

Marine National Facility

Australian Government

Commissioned on 12 December 2014, *RV Investigator* is capable of delivering up to **300 research days each year**

Length **93.9 m**

Beam (width) **18.5 m**

Height (waterline to top of ship) **37 m**

Draft (waterline to bottom of ship) **6.2 m**

Gross tonnage **6082 t**

10 Internal stories

12 On board laboratories

20 Ship crew

10 CSIRO staff (approx.)

30 Science participants

60 Days endurance

10,000 Nautical miles range

Overview

1. OpenRVDAS Introduction/Summary
2. RVIDAS – The Infrastructure Around OpenRVDAS
3. Novel Additions
   - Jinja Templates
   - RegEx Device Definitions
   - TimescaleDB Database Connector

CSIRO Marine National Facility

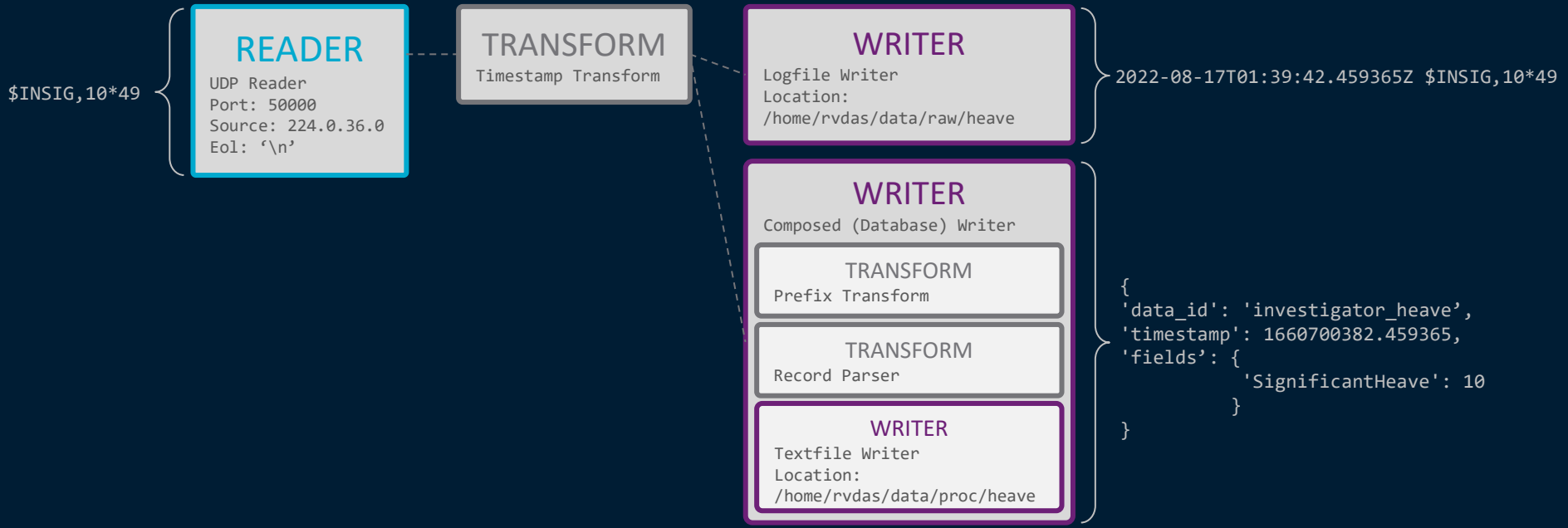
OceanDataTools/ **openrvdas**

An open source data acquisition system designed for use on research vessels and other scientific installations

15 Contributors | 31 Issues | 32 Stars | 15 Forks

What Does OpenRVDAS Look Like on the RV Investigator?

Loggers


$INSIG,10*49
READER
UDP Reader
Port: 50000
Source: 224.0.36.0
Eol: '\n'
TRANSFORM
Timestamp Transform
WRITER
Logfile Writer
Location:
/home/rvdas/data/raw/heave
2022-08-17T01:39:42.459365Z $INSIG,10*49
WRITER
Composed (Database) Writer
TRANSFORM
Prefix Transform
TRANSFORM
Record Parser
WRITER
Textfile Writer
Location:
/home/rvdas/data/proc/heave
{
'data_id': 'investigator_heave',
'timestamp': 1660700382.459365,
'fields': {
'SignificantHeave': 10
}
}

Voyage Config

```
cruise:
  id: 'IN2023_V03'
  start: '2023-05-09'
  end: '2023-05-16'

loggers:
    heave_aframe:
      configs:
      - heave_aframe->off
      - heave_aframe->file/net
      - heave_aframe->file/net/db

modes:
  'off':
    heave_aframe: heave_aframe->off
 'file/net':
    heave_aframe: heave_aframe-> file/net
 'file/net/db':
    heave_aframe: heave_aframe->file/net/db

 heave_active_comp_aframe->off:
    name: heave_active_comp_aframe->off
     readers:
        - class: UDPReader
            kwargs:
              port: 50201
              source: 224.0.36.0
    transforms:
      - class: TimestampTransform
    writers:
      - class: LogfileWriter
      - class: ComposedWriter
          kwargs:
              transforms:
                - class: PrefixTransform
                - class: RegexParseTransform
           writers:
                - class: TextFileWriter  # write parsed to logfile
                  kwargs:
                    filename: /data/proc/IN2023_V03/heave_active_comp_aframe
                    split_by_date: True
heave_active_comp_aframe->file/net/db:
    name: heave_active_comp_aframe->file/net/db
      readers:
        - class: UDPReader
          kwargs:
            port: 50201
            source: 224.0.36.0
    transforms:
      - class: TimestampTransform
    writers:
      - class: LogfileWriter
        kwargs:
          filebase: /data/raw/heave_active_comp_aframe
      - class: ComposedWriter
        kwargs:
          transforms:
            - class: PrefixTransform
            - class: RegexParseTransform
          writers:
            - class: TextFileWriter  # write parsed to logfile
              kwargs:
                filename: /data/proc/IN2023_V03/heave_active_comp_aframe
            - class: DatabaseWriter
```

Ship Devices

```
 investigator_heave_aframe:
 category: "device"
    device_type: "Heave"
    description: "Heave for aframe"

    fields:
      HeaveAcceleration: "HeaveAcceleration"
      HeavePosition: "HeavePosition"
      HeaveVelocity: "HeaveVelocity"
      SignificantHeave: "SignificantHeave"

investigator_heave_ctdboom:
    category: "device"
    device_type: "Heave"
    description: "Heave for ctdboom"

    fields:
      HeaveAcceleration: "HeaveAcceleration"
      HeavePosition: "HeavePosition"
      HeaveVelocity: "HeaveVelocity"
      SignificantHeave: "SignificantHeave"
```

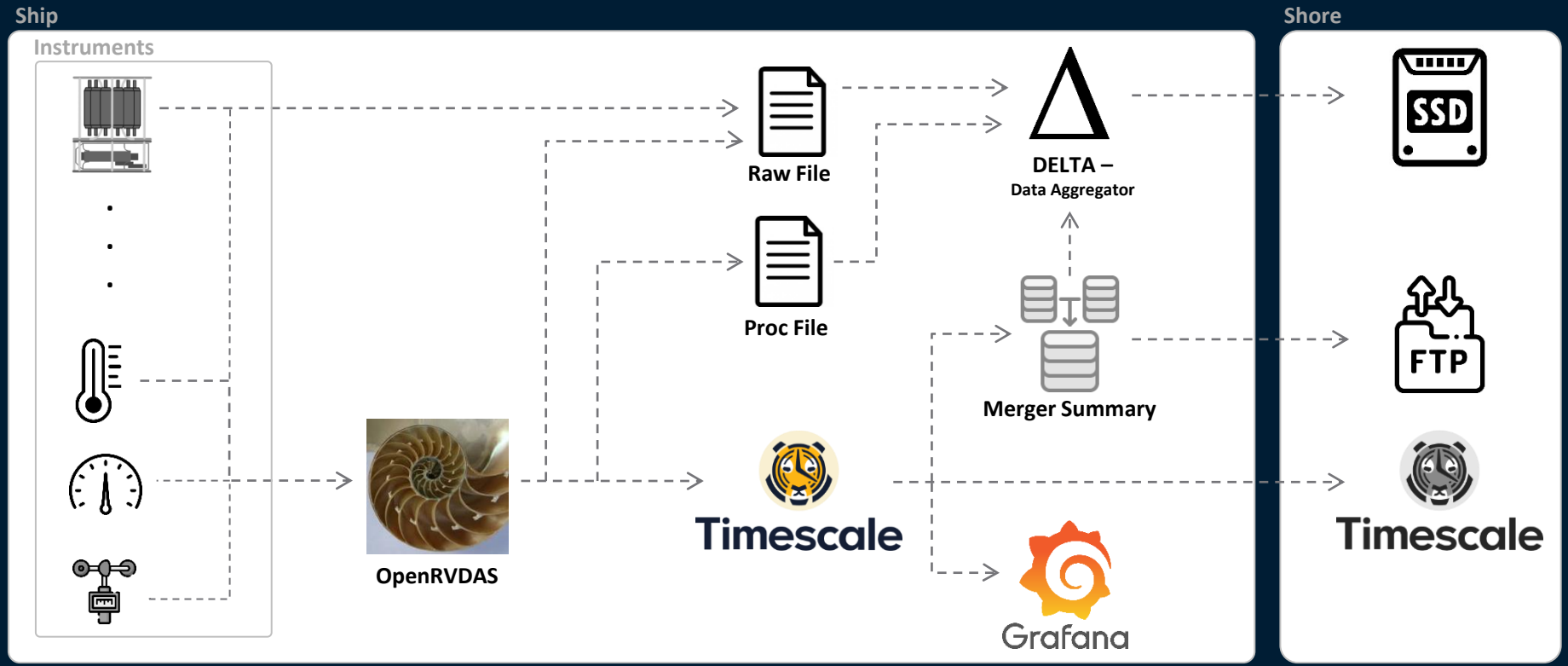
Device Definition

```
Heave:
  category: "device_type"
  instrument: "Heave"
  description: "Heave for Aframe and CTD Boom"

  format:
      - $INSIG,{SignificantHeave:of}*{Checksum:x}
        #e.g $INSIG,10.2*49

  fields:
    SignificantHeave:
      units: "(m)"
      instrument: "Heave"
      description: "Significant Heave”
```

Config files Summary

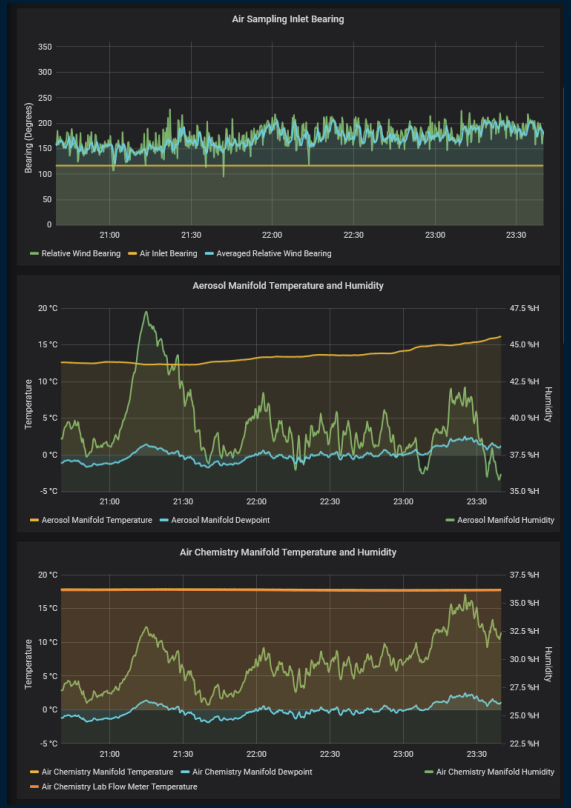
Voyage Config	Ship Devices	Device Definitions
Define voyage name, dates, modes and loggers.	Match device definition to voyage config (can have multiple of the same type of device).	Define the structure of incoming data.
IN2023_V01.yaml 	investigator_devices.yaml 	kongsberg_dps112.yaml 
Length of a Voyage Config File: 5445 Lines	Total Instruments in Ship Devices: 70	Number of Device Definitions: 45

RV Investigator Data Acquisition System (RVIDAS)



Grafana

- Fantastic data visualisation tool
- Plug in architecture - many extensions
- Intuitive
- Can set up multiple data sources (i.e. databases)
- Look at data sources without creating data copies
- Create dashboards based on deployments, locations, teams
- Compare primary and secondary sensors
- Open Soruce – Free!



CSIRO Marine National Facility

Jinja Template

Jinja Template Engine

```
loggers:
  {% for device in devices %}
  {{ device }}:
    configs:
    - {{ device }}->off
    - {{ device }}->file/net
    - {{ device }}->file/net/db
  {% endfor %}
```

```
loggers:
    heave_aframe:
      configs:
      - heave_aframe->off
      - heave_aframe->file/net
      - heave_aframe->file/net/db
    heave_ctdboom:
      configs:
      - heave_ctdboom->off
      - heave_ctdboom->file/net
      - heave_ctdboom->file/net/db
    underway_seawater:
      configs:
      - underway_seawater->off
      - underway_seawater->file/net
      - underway_seawater->file/net/db
    airsamplinginlet:
      configs:
      - airsamplinginlet->off
      - airsamplinginlet->file/net
      - airsamplinginlet->file/net/db
```

More Config Files?!

Jinja Template	Voyage Devices	Voyage Config
Control the logic of converting Voyage Devices to Voyage Config.	Define voyage name, dates, device ports and special cases.	Define voyage name, dates, modes and loggers.
voyage_template.jinja 	IN2023_V01_devices.yaml 	IN2023_V01.yaml 
Only changes when new methods are being used (e.g. transforms)	Total Instruments in Ship Devices: 236 Lines	Length of a Voyage Config File: 5445 Lines

Voyage Devices vs Voyage Config

```yaml
cruise:
  id: 'IN2023_V03'
  start: '2023-05-09'
  end: '2023-05-16'

defaults:
  sourceip: '224.0.36.0'
  definition_path: /local/investigator/devices/investigator_devices.yaml
  file_pathbase: /mnt/rvidas_data/file
  prefix: investigator

 heave_aframe:
    port: 50201
```

```yaml
cruise:
  id: 'IN2023_V03'
  start: '2023-05-09'
  end: '2023-05-16'

loggers:
    heave_aframe:
      configs:
      - heave_aframe->off
      - heave_aframe->file/net
      - heave_aframe->file/net/db

modes:
  'off':
    heave_aframe: heave_aframe->off
 'file/net':
    heave_aframe: heave_aframe-> file/net
 'file/net/db':
    heave_aframe: heave_aframe->file/net/db

 heave_active_comp_aframe->off:
    name: heave_active_comp_aframe->off
     readers:
        - class: UDPReader
          kwargs:
            port: 50201
            source: 224.0.36.0
    transforms:
      - class: TimestampTransform
    writers:
      - class: LogfileWriter
      - class: ComposedWriter
        kwargs:
          transforms:
            - class: PrefixTransform
            - class: RegexParseTransform
       writers:
            - class: TextFileWriter  # write parsed to logfile
              kwargs:
                filename: /data/proc/IN2023_V03/heave_active_comp_aframe
                split_by_date: True
heave_active_comp_aframe->file/net/db:
    name: heave_active_comp_aframe->file/net/db
      readers:
        - class: UDPReader
          kwargs:
            port: 50201
            source: 224.0.36.0
    transforms:
      - class: TimestampTransform
    writers:
      - class: LogfileWriter
        kwargs:
          filebase: /data/raw/heave_active_comp_aframe
      - class: ComposedWriter
        kwargs:
          transforms:
            - class: PrefixTransform
            - class: RegexParseTransform
          writers:
            - class: TextFileWriter  # write parsed to logfile
              kwargs:
                filename: /data/proc/IN2023_V03/heave_active_comp_aframe
            - class: DatabaseWriter
```

RegEx Device Definitions

Device Definition

```
Heave:
  category: "device_type"
  instrument: "Heave"
  description: "Heave for Aframe and CTD Boom"

  format:
      - $INSIG,{SignificantHeave:of}*{Checksum:x}
        #e.g $INSIG,10.2*49

  fields:
    SignificantHeave:
      units: "(m)"
      instrument: "Heave"
      description: "Significant Heave”
```

PyParse

```
$INSIG,{SignificantHeave:of}*{Checksum:x}
```

```
$INSIG,10.2*49

Significant Heave: 10.2 (float)
      Checksum: 49 (int)
```

RegEx

```
\$INSIG,(?P<SignificantHeave>\-?\d*\.?\d*)\s*\*(?P<Checksum>[\da-fA-F]{2})
```

```
$INSIG,10.2*49

Significant Heave: "10.2" (string)
      Checksum: "49" (string)
```

Device Definition

```
Heave:
  category: "device_type"
  instrument: "Heave"
  description: "Heave for Aframe and CTD Boom"

  format:
      - \$INSIG,(?P<SignificantHeave>\-?\d*\.?\d*)\s*\*(?P<Checksum>[\da-fA-F]{2})
        #e.g $INSIG,10.2*49

  fields:
    SignificantHeave:
      units: "(m)"
      instrument: "Heave"
      description: "Significant Heave"
      data_type: float
```

Utilising Superfunctions

```
class RegexRecordParser(RecordParser):
    def __init__(self, record_format=None, field_patterns=None, metadata=None,
    definition_path=DEFAULT_DEFINITION_PATH, return_das_record=False, return_json=False,
    metadata_interval=None, quiet=False, prepend_data_id=False, delimiter=':'):

        super().__init__(record_format=record_format, field_patterns=field_patterns, metadata=metadata,
        definition_path=definition_path, return_das_record=return_das_record, return_json=return_json,
        metadata_interval=metadata_interval, quiet=quiet, prepend_data_id=prepend_data_id,
        delimiter=delimiter)

        self.data_type_map = {}
        < build dictionary that maps data formats >
```

```
def parse_record(self, record):
    parsed_record = RecordParser.parse_record(self, record)
    if parsed_record:
        < use 'data_type_map' to cast the data type of each value in the record >
    return parsed_record
```

```
def _parse_field_string(self, field_string, compiled_field_patterns)

def parse_for_data_id(self, data_id, field_string)
```

TimescaleDB Database Connector

Database Choices

Relational Databases

Relational databases can fall down under the heavy insert workload of time series data.

NoSQL Databases

NoSQL databases sacrifice query complexity and a well known query language

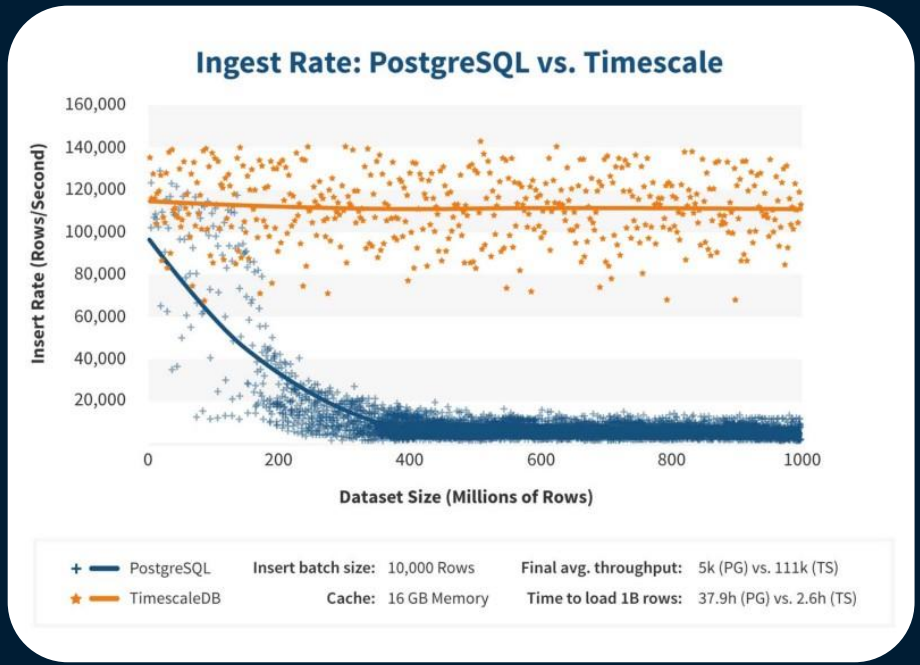
CSIRO Marine National Facility

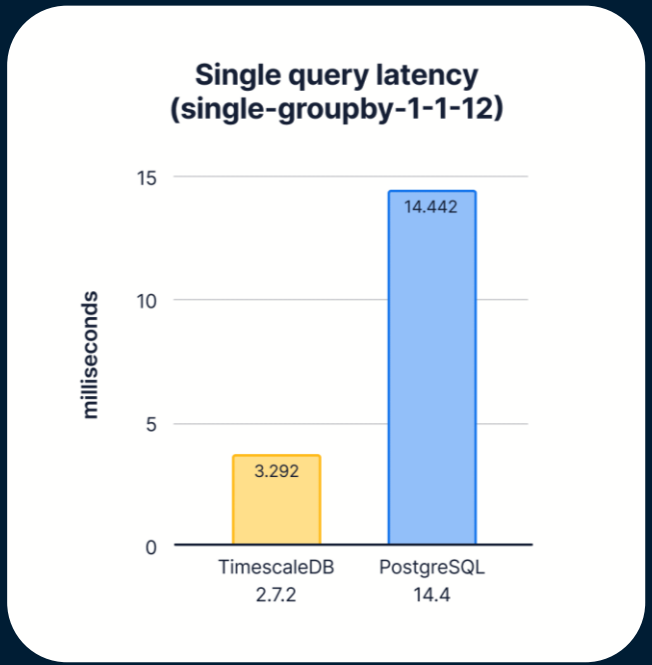
timescale/ **timescaledb**

An open-source time-series SQL database optimized for fast ingest and complex queries. Packaged as a PostgreSQL extension.

83	1	15k	793
Contributors	Used by	Stars	Forks

Benchmarks

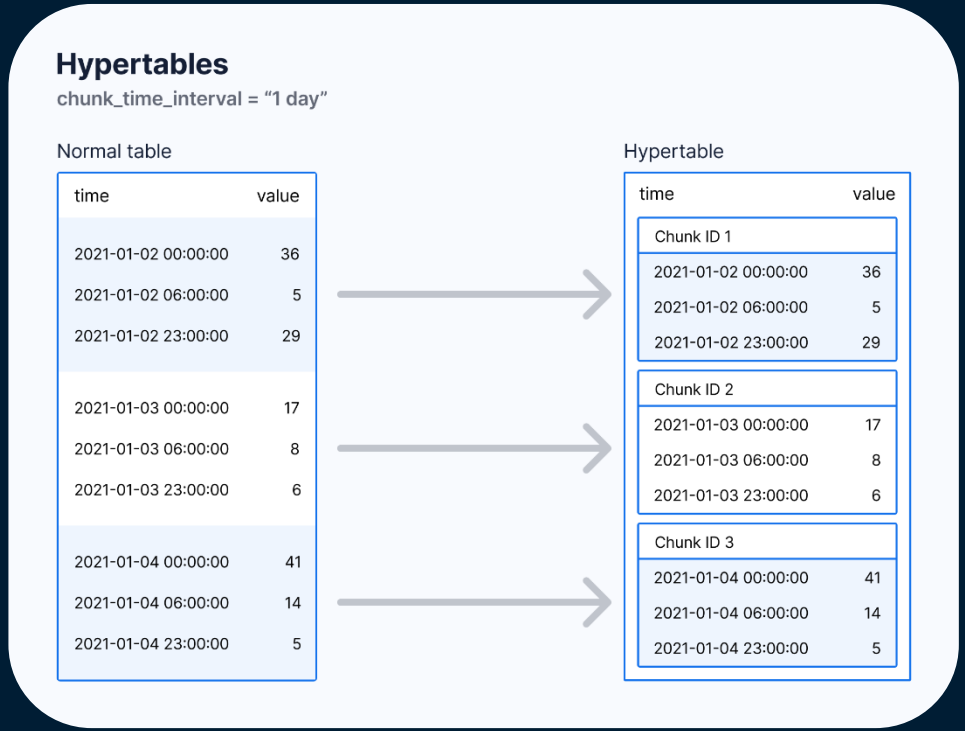




https://www.timescale.com/blog/postgresql-timescaledb-1000x-faster-queries-90-data-compression-and-much-more/
https://www.timescale.com/blog/timescaledb-vs-6a696248104e/

Hypertables

- Hyper table: partitioning of data, while presenting the abstraction of a single, virtual table.
- This partitioning enables faster queries by quickly excluding irrelevant data, as well as enabling enhancements to the query planner and execution process.
- Chunk: child tables that create a hypertable



https://www.timescale.com/blog/recap-getting-started-with-timescaledb-and-defining-best-practices/

Database Connector

- Psycopg2
  - Popular
  - Reliable
  - Thread safety [1]
- Wide table format
  - In PostgreSQL NULLs take trivial space [2]
- Automatic table and column insertion
  - New data streams only need to be defined in the device definitions.
  - Done via exception handling with Psycopg

[1] https://www.psycopg.org/docs/
[2] https://www.postgresql.org/docs/14/storage-page-layout.html#STORAGE-TUPLE-LAYOUT

CSIRO Marine National Facility

	id	timestamp	concentration	flowrate	error	unitid	pressure	temp	volflow	massflow	setpoint	gas
1	1369460	2023-05-31 05:07:39.749340 +00:00	<null>	<null>	<null>	B	959	23.67	1.006	0.956	1.006	Air
2	1369459	2023-05-31 05:07:39.743742 +00:00	1888.269	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
3	1369458	2023-05-31 05:07:38.761862 +00:00	<null>	<null>	<null>	B	959	23.67	1.006	0.957	1.006	Air
4	1369457	2023-05-31 05:07:38.757284 +00:00	1873.465	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
5	1369456	2023-05-31 05:07:37.803346 +00:00	<null>	<null>	<null>	B	959	23.67	1.005	0.956	1.006	Air
6	1369455	2023-05-31 05:07:37.797195 +00:00	1817.715	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
7	1369454	2023-05-31 05:07:36.747985 +00:00	<null>	<null>	<null>	B	958	23.67	1.006	0.956	1.006	Air
8	1369453	2023-05-31 05:07:36.742317 +00:00	1857.535	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
9	1369452	2023-05-31 05:07:35.744501 +00:00	<null>	<null>	<null>	B	960	23.67	1.006	0.957	1.006	Air
10	1369451	2023-05-31 05:07:35.738821 +00:00	1849.544	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
11	1369450	2023-05-31 05:07:34.791200 +00:00	<null>	<null>	<null>	B	960	23.67	1.005	0.957	1.006	Air
12	1369449	2023-05-31 05:07:34.786769 +00:00	1826.127	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
13	1369448	2023-05-31 05:07:33.832550 +00:00	<null>	<null>	<null>	B	959	23.67	1.006	0.957	1.006	Air
14	1369447	2023-05-31 05:07:33.826086 +00:00	1796.358	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
15	1369446	2023-05-31 05:07:32.743500 +00:00	<null>	<null>	<null>	B	958	23.67	1.006	0.955	1.006	Air
16	1369445	2023-05-31 05:07:32.737394 +00:00	1796.971	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
17	1369444	2023-05-31 05:07:31.743911 +00:00	<null>	<null>	<null>	B	960	23.67	1.006	0.957	1.006	Air
18	1369443	2023-05-31 05:07:31.737257 +00:00	1818.458	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
19	1369442	2023-05-31 05:07:30.744445 +00:00	<null>	<null>	<null>	B	960	23.67	1.005	0.957	1.006	Air
20	1369441	2023-05-31 05:07:30.740380 +00:00	1841.41	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
21	1369440	2023-05-31 05:07:29.793206 +00:00	<null>	<null>	<null>	B	960	23.67	1.006	0.957	1.006	Air
22	1369439	2023-05-31 05:07:29.787594 +00:00	1838.926	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
23	1369438	2023-05-31 05:07:28.748826 +00:00	<null>	<null>	<null>	B	960	23.67	1.005	0.957	1.006	Air
24	1369437	2023-05-31 05:07:28.743415 +00:00	1835.084	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
25	1369436	2023-05-31 05:07:27.745251 +00:00	<null>	<null>	<null>	B	960	23.67	1.006	0.957	1.006	Air
26	1369435	2023-05-31 05:07:27.738479 +00:00	1872.803	1.006	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
27	1369434	2023-05-31 05:07:26.746703 +00:00	<null>	<null>	<null>	B	960	23.67	1.005	0.957	1.006	Air
28	1369433	2023-05-31 05:07:26.739440 +00:00	1829.291	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>
29	1369432	2023-05-31 05:07:25.742096 +00:00	<null>	<null>	<null>	B	959	23.67	1.005	0.955	1.006	Air
30	1369431	2023-05-31 05:07:25.736614 +00:00	1843.408	1.005	0x00	<null>	<null>	<null>	<null>	<null>	<null>	<null>

Work to be done

- OpenRVDAS Transforms for data conversions (i.e. changing units, wind vectors, etc.)
- OpenRVDAS UDP Broadcasts
- Timescale ship to shore database replication
- Underway merger (Timescale Real-time Views, exported to csv and NetCDF)
- Database: Stacking PostgreSQL extensions – postGIS

Thank you

Engineering & Technology
Ella Pietraroia
Software Engineer / Senior Technical Officer

ella.pietraroia@csiro.au

CSIRO acknowledges the Traditional Owners of the lands on which we live and work across Australia, and pays its respect to Elders past and present.