University of Tasmania
GPO Box 367
Hobart 7001
Australia

imos.org.au

April 2026

IMOS NESP 5.9

Dugong Aerial Survey Optimised Data

Product Technical Document

Version 1.1

Karishma Khanna
Australian Ocean Data Network AODN / IMOS

Australia's Integrated Marine Observing System (IMOS) is enabled by the National Collaborative Research Infrastructure Strategy (NCRIS). It is operated by a consortium of institutions as an unincorporated joint venture, with the University of Tasmania as Lead Agent.

IMOS acknowledges the Traditional Custodians and Elders of the land and sea on which we work and observe, and recognise them as Australia's first marine scientists and carers of sea Country. We pay our respects to Aboriginal and Torres Strait Islander peoples past and present.

1. Version History

Version	Date	Comments	Author
v1.0	March 2026	Initial Release	Karishma, K
v1.1	April 2026	Minor Revision	Galindo, T

Citation

Karishma, K. (2026). Dugong Aerial Survey Optimised Data Product Technical Document. Version 1.1. Integrated Marine Observing System.

Copyright/Creative Commons Licence

Table of Contents

2. Background

2.1 Overview

The Dugong data pipeline consolidates historical and ongoing survey data from multiple distinct sources into a single, unified, analytics-ready Parquet dataset.

The raw deliveries are provided as a mix of historical Data Management System (DMS) records and James Cook University (JCU) geospatial shapefile archives. This work converts those heterogeneous deliveries into a consistent dataset aligned to a standardised Darwin Core-style occurrence structure with:

- cleaned and standardised coordinates
- validated event dates
- exploded biological counts into adult and calf occurrence records
- derived `eventID` and `occurrenceID` keys
- an H3 spatial index for fast spatial grouping and joins
- Australian Marine Region spatial tags
- a separate rejected-records output with explicit rejection reasons

The workflow is implemented as an Extract → Transform → Load pipeline orchestrated using Prefect.

Basis of the Dugong Aggregated Data Product

The goal of this aggregated data product is to consolidate Dugong aerial survey data from multiple delivery formats into a single analytics-ready dataset that is:

- interoperable with existing Darwin Core tooling and systems
- enriched with spatial indices for efficient geospatial queries
- traceable back to original source records via stable identifiers

2.2 Source Data Characteristics

The Dugong Data Product combines data from two primary source systems:

- **DMS data** delivered as Parquet
- **JCU data** delivered as zipped Shapefile archives

DMS Data

The DMS source is delivered as a Parquet file containing historically aggregated dugong sightings. Typical source fields include:

- survey identifier
- sighting identifier
- survey date/time
- latitude and longitude
- transect/block identifiers
- total group count
- calf count
- observer and survey condition attributes

These grouped biological counts are expanded during processing into separate life-stage occurrence records.

JCU Data

The JCU source is delivered as ZIP archives containing Shapefiles. These archives may contain:

- a **Sightings** shapefile
- a **Herd** shapefile
- or both

The processing logic treats the **Sightings** shapefile as required:

- if sightings data exists, it is processed
- if herd data also exists, herd data is processed and combined with sightings
- if sightings data does not exist, the archive is skipped

This reflects the implementation in the ETL code, where herd data is supplementary and only consumed when sightings data is present.

Common Data Characteristics

Across both sources:

- dugong counts are represented as grouped counts and must be expanded into adult and calf records
- coordinates may be missing in raw source files
- dates may be missing in some source records
- duplicate records may occur after source consolidation
- source schemas differ and must be standardised before downstream enrichment

2.3 Output Data Products

Two outputs are generated per pipeline execution.

Validated dataset (Master Parquet)

A single unified Parquet file is produced containing:

- validated, standardised occurrence records
- exploded life-stage rows (`adult` , `calves`)
- `h3Index`
- `australianMarineRegionsTags`

Rejected dataset (Parquet)

Records that fail validation are written separately:

- written as a single rejected Parquet output
- includes `RejectionReason`
- used for traceability and diagnostics to ensure zero data loss during extraction
- preserves native types for stable fields (e.g. dates, coordinates, counts) while stringifying highly variable/raw-source fields to avoid schema merge conflicts between heterogeneous shapefile structures

3. Methodology

The Dugong Aggregated Data Product relies on the standard Extract, Transform, and Load (ETL) methodology orchestrated with Prefect.

3.1 Extract

Objective: read DMS and JCU source files from S3, validate required fields, standardise them to a common schema, and expand grouped counts into occurrence-style records.

A) DMS extract

For the DMS source, the pipeline:

1. reads the Parquet file from S3 into memory
2. validates required fields:
    - coordinates
    - event date
3. separates rejected records with reasons:
    - `Missing Coordinates`
    - `Missing Date`
4. explodes grouped counts into:
    - `adult`
    - `calves`
5. derives identifiers:
    - `eventID`
    - `occurrenceID`
6. maps source fields into the standard Dugong extract schema

B) JCU extract

For the JCU source, the pipeline:

1. reads the ZIP archive from S3 into memory
2. extracts Shapefiles to a temporary directory
3. loads Shapefiles using `pyogrio`
4. identifies:
    - the sightings table
    - the herd table
5. applies the following processing logic:
    - if sightings exists, process sightings
    - if herd also exists, process herd and combine results
    - if sightings does not exist, skip the archive
6. validates required fields:
    - coordinates

    - event date
7. derives identifiers:
    - `eventID`
    - `occurrenceID`
8. explodes grouped counts into:
    - `adult`
    - `calves`
9. maps fields into the standard Dugong extract schema

C) Identifier derivation

The pipeline derives stable identifiers during extract to preserve traceability back to the original source records.

DMS

- `eventID` is taken directly from the source survey identifier: `surv_id`
- `occurrenceID` is constructed as:

`{eventID}|{sighting_id}|{organismQuantityType}`

For example, a single DMS sighting may produce two occurrence rows after explode:

- one for `adult`
- one for `calves`

This means the same source sighting can generate two distinct `occurrenceID` values.

For rejected DMS rows, where a full life-stage split has not occurred, the pipeline uses a fallback occurrence identifier structure to retain traceability to the original sighting.

JCU

- `eventID` is derived from the survey region and observation date using:

`{region_abbr}-{YYYY}-{MM}`

The `region_abbr` value is standardised from the raw `region` field using the following mapping:

- `northern great barrier reef` → `nGBR`
- `central great barrier reef` → `cGBR`
- `southern great barrier reef` → `sGBR`
- `gulf of carpentaria` → `GoC`
- `torres strait` → `TS`

- `moreton bay` → `MB`
- `hervey bay` → `HB`
- `shark bay` → `SB`
- `exmouth` → `Ex`

If a region value does not match one of the configured mappings, it is assigned the default value:

- `Unknown`

- `occurrenceID` is constructed as:

`{eventID}|{recordNumber}|{organismQuantityType}`

where:

- `recordNumber` is the original source sighting identifier
- `organismQuantityType` is either `adult` or `calves`

For JCU sightings, `recordNumber` is derived from `sighting_0`.
For JCU herd data, `recordNumber` is derived from `sighting_I`.

This ensures each exploded life-stage record remains unique while preserving linkage back to the original source observation.

D) Extract schema validation

The extracted DMS and JCU records are validated against `extract.schema.json` using `validate_extract_schema()`.

This validation step:

- reorders columns to the configured schema definition
- checks Arrow type compatibility
- ensures both sources conform to the same extracted dataset structure before transform

---

3.2 Transform

Objective: validate coordinates, remove duplicates, and enrich the standardised dataset with spatial attributes.

Coordinate validation

Coordinates are validated to ensure they fall within global bounds:

- `-90 <= decimalLatitude <= 90`
- `-180 <= decimalLongitude <= 180`

Rows outside these ranges are rejected with:

- `Invalid Coordinates (out of bounds)`

Duplicate detection

Duplicate records are identified using a full-row distinct comparison.
The first occurrence is retained and subsequent duplicates are rejected with:

- `Duplicate Record`

Geospatial Enrichment

Valid records are enriched with:

- `h3Index` using configured H3 resolution (default: 4, ~1,770 km^2 hexagons)
- `australianMarineRegionsTags` using Australian Marine Region spatial tagging

The H3 spatial index enables efficient geospatial queries and aggregations without requiring complex GIS operations.

3.3 Load

Objective: persist validated and rejected DataFrames to S3 as Parquet files for downstream consumption and traceability.

Parquet serialization

Both the validated master dataset and rejected records are serialized to Parquet format using Polars:

1. the DataFrame is written to an in-memory buffer using `write_parquet()`
2. the buffer contents are uploaded to S3 using the configured Prefect block

Valid records upload

Valid records are written to the optimised data bucket:

- bucket: `optimised-bucket`
- path: `dugong/{filename}.parquet`

If the valid DataFrame is empty, the upload is skipped with a warning.

Rejected records upload

Rejected records from both extract and transform stages are concatenated and written to the error bucket:

- bucket: `error-bucket`
- path: `dugong/{filename}_rejected.parquet`

If no rejected records exist, the upload is skipped.

Error handling

The load stage implements explicit error handling:

- serialization failures raise `ValueError` with context
- S3 upload failures raise `IOError` with the target path
- all operations are logged with byte sizes for traceability

3.4 Validation Rules

Validation is applied across extract and transform stages. Each rejected set is preserved and labelled.

1. **Missing coordinates**
    - rule: latitude and longitude must be non-null
    - reason: `Missing Coordinates`
2. **Missing date**
    - rule: event date field must be non-null
    - reason: `Missing Date`
3. **Invalid global coordinate ranges**
    - rule: `-90 <= decimalLatitude <= 90` and `-180 <= decimalLongitude <= 180`
    - reason: `Invalid Coordinates (out of bounds)`
4. **Duplicate detection**
    - rule: full-row first-distinct check
    - reason: `Duplicate Record`

Rejected rows from all stages are concatenated into a single rejected DataFrame and written to the error bucket.

3.5 Standardisation and Schema Validation

A strict extracted schema is defined in `extract.schema.json` and enforced during the extract stage.

Before transform:

- both DMS and JCU outputs are mapped into the same column structure
- columns are ordered consistently
- extracted records are validated against the configured Arrow schema

This ensures stable typing and a consistent structure before downstream transform and load steps.

3.6 Load Strategy

Valid output

The validated master dataset is written as a Parquet file to the optimised bucket:

- `dugong/dugong.parquet`

Rejected output

Rejected records from extract and transform stages are concatenated and written as a Parquet file to the error bucket:

- `dugong/dugong_rejected.parquet`

3.7 Output Schema (Summary)

Validated dataset

The validated Dugong dataset includes the following fields:

name	type	nullable	description
eventID	string	**false**	Survey identifier (DMS: surv_id, JCU: region-YYYY-MM)
occurrenceID	string	**false**	Unique occurrence identifier
recordNumber	int64	true	Original source sighting identifier
occurrenceType	string	true	Type of sighting (e.g., Primary, Incidental)
eventDate	date	**false**	Survey observation date
decimalLatitude	float64	**false**	Latitude in decimal degrees
decimalLongitude	float64	**false**	Longitude in decimal degrees
locationID	int64	true	Transect identifier
locationName	string	true	Block/region name
organismQuantityType	string	**false**	Life stage: `adult` or `calves`
organismQuantity	int64	**false**	Count of individuals
beaufortScale	int8	true	Sea state observation (0-12)
turbidityScale	int8	true	Water turbidity observation
recordedBy	string	true	Observer name
scientificName	string	**false**	Species name: `Dugong dugon`
scientificNameID	string	**false**	WoRMS LSID: `urn:lsid:marinespecies.org:taxname:220227`

h3Index	string	**false**	H3 spatial index (resolution 4)
australianMarineRegionsTags	string	true	Pipe-separated marine region tags

Rejected dataset

The rejected output includes all mapped source fields plus:

name	type	nullable	description
RejectionReason	string	**false**	Reason for rejection (e.g., `Missing Coordinates`, `Duplicate Record`)

3.8 Configuration

The pipeline is configuration-driven using JSON files in the `config/` directory.

Source configuration

- `dms.pyarrow_table_factory.json`
  defines the DMS Parquet source block and path
- `jcu_shapefile.file_interface.json`
  defines the JCU archive block and list of ZIP files to ingest

Schema configuration

- `extract.schema.json`
  defines the standardised extracted schema and datatypes

3.9 Dependencies

Key libraries used in this pipeline include:

- `polars` for DataFrame operations and transformations
- `pyogrio` for Shapefile reading
- `pyarrow` for schema validation and Parquet I/O
- `polars_h3` for H3 spatial indexing
- `prefect` for workflow orchestration
- `pydantic` for configuration validation

3.10 Running the Pipeline

Prerequisites

1. AWS credentials configured for S3 access
2. Prefect environment available (optional for orchestration/monitoring)

Execution

```
cd src/datauplift/dugong
python -m datauplift.dugong.etl
```

4. Flow Advantages

Reproducibility the original Dugong data sources (DMS Parquet and JCU Shapefiles) require specialised geospatial software and manual processing. The ETL procedure makes this process reliable and replicable.

Space Efficiency heterogeneous source data is consolidated into a single optimised Parquet file (~500KB with modern compression), optimising storage and access.

Time Efficiency scheduled updates allow users to access up-to-date aggregated data without re-implementing the extraction and transformation steps.

Data Quality built-in validation rules ensure coordinate integrity, date validity, and duplicate detection with full traceability of rejected records.

Cloud-native workflows reduce I/O overhead and enable seamless integration with AWS S3 storage.